République **A**lgérienne **D**émocratique et **P**opulaire
Ministère de l'**E**nseignement **S**upérieur et de la **R**echerche **S**cientifique

UNIVERSITE **M**OULOUD **M**AMMERI DE **T**IZI **O**UZOU

FACULTE DU **G**ENIE **E**LECTRIQUE ET D'**I**NFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE

Mémoire de Fin d'Etudes de MASTER ACADIMIQUE

Domaine : **Sciences et Technologies**

Filière : **Génie électrique**

Spécialité : **Electronique biomédicale**

Présenté par

Redouane LEKHAL

Thème

Application des SVM pour la reconnaissance d'extrasystoles.

Mémoire soutenu publiquement le 15/06/2015 devant le jury composé de :

M^{me} Z . AMEUR
Professeur, Université MOULOUD MAMMERI de Tizi Ouzou, Président

M^{me} Z . AMIROU
Maître de conférences classe « A », Université MOULOUD MAMMERI de Tizi Ouzou, Encadreur.

M^{r} M. LAHDIR
Maître de conférences classe « A », Université MOULOUD MAMMERI de Tizi Ouzou, Examinateur.

M^{r} M. SEHAD
Maître de conférences classe « B », Université MOULOUD MAMMERI de Tizi Ouzou, Examinateur.

Remerciements

Remerciements

J'exprime ma profonde gratitude et mes sincères remerciements à ma promotrice, M^{me} AMIROU Zahia, Maître de conférences classe (A) à l'Université Mouloud Mammeri de Tizi Ouzou, pour son sérieux, sa rigueur, et son aide qui n'ont d'égale que ses connaissances et sa passion pour son métier.

Je tien à remercier chaleureusement le président ainsi que tous les membres du jury, d'avoir honoré cette soutenance.

Je remercie également toute personne m'ayant encouragé tout au long de l'accomplissement de ce travail, qu'ils retrouvent ici l'expression de ma profonde reconnaissance.

Résumé

L'objectif de ce travail est de reconnaitre des battements cardiaques pathologiques (extrasystoles) sur des signaux ECG en effectuant une classification de ces battements en deux catégories : normal et pathologique.

Pour ce faire, nous avons choisi d'utiliser un algorithme nommé Support Vector Machines (SVM). Les données utilisées dans cette application sont des enregistrements ECG issues de la base de données internationale MIT-BIH (Massachusetts Institute of Technologie/Beth Israel Hospital) arrhythmia data-base. Chaque battement de l'enregistrement (échantillon) est caractérisé par un vecteur de caractéristiques (attributs) et associé à une des deux catégories. La caractérisation consiste à localiser le pic R de chaque battement et d'ouvrir une fenêtre autour de ce pic pour l'extraction des différentes caractéristiques caractérisant les deux types de battements. Nous avons construit une base d'apprentissage et une base de test de différent enregistrement. La sélection des hyper-paramètres se fait par une technique de validation croisée, les hyper-paramètres choisis sont ceux pour lequel le taux de bonne classification est maximal. Les résultats obtenus montrent que les SVMs sont des techniques très performantes et que leur pouvoir de généralisation s'améliore en choisissant une base d'apprentissage variante (très riche en information).

Mots-clés : Apprentissage statistique, Classification supervisé, Support Vector Machines, Classification par SVM, Caractérisation des battements cardiaques.

Abréviations

Abréviations

ACP	**Analyse en Composante Principale.**
DSP	**Densité Spectral de Puissance.**
ECD	**Extraction des Connaissances à partir des Données**
ECG	**ElectroCardioGramme.**
ESV	**ExtraSystole Ventriculaire.**
kNN	**(k Nearest Neighbor),**
KDD	**Knowledge Discovery from Data**
LLSF	**(Linear Least Square Fit).**
MIT-BIH	**Massachusetts Institute of Technologie/Beth Israel Hospital.**
MRE	**Minimisation de Risque Empirique.**
MRS	**Minimisation du Risque Structurel.**
PVC	**Contraction Ventriculaire Primaturée.**
SMO	**Séquentiel Minimisation Optimisation.**
SV	**Support Vector.**
SVM	**Support Vector Machine.**
VC	**dimention de Vapnik et Chervonenkis.**

Table des figures

Table des figures

Liste des tableaux

Liste des tableaux

Sommaire

Sommaire

Chapitre II : Support Vector Machines

Chapitre III : caractérisation des battements cardiaques

Chapitre IV : Résultats de la classification

Introduction Générale

Introduction générale

L'évolution de la technologie a bouleversé le monde et a facilité notre vie quotidienne. Aujourd'hui, les tâches les plus difficiles sont réalisées par des machines. Cette évolution a incité les chercheurs dans différents domaines à apporter des solutions aux problèmes que posent ces machines, vu qu'elles nécessitent parfois l'intervention humaine. Les chercheurs tentent alors à programmer des machines capables d'effectuer des tâches qui requirent de l'intelligence.

Cependant, programmer des machines capables de s'adapter à toutes les situations et éventuellement d'évoluer en fonction de nouvelles contraintes est difficile. L'enjeu est de contourner cette difficulté en dotant la machine de capacités d'apprentissage lui permettant de tirer parti de son expérience. C'est pourquoi des recherches sur le raisonnement automatique et sur l'apprentissage se sont développés. Les domaines d'application de l'apprentissage artificielle sont nombreux : fouille de données (FD), bioinformatique, génie des procédés, aide au diagnostic médical, télécommunications, interface cerveau-machines, et bien d'autres.

La fouille de données permet l'extraction Automatique d'informations prédictives à partir de gros volumes de données. Elle représente un mélange d'outils provenant de la Statistique, de l'Intelligence Artificielle et de l'Informatique (reconnaissance de formes).

Le travail qui nous a été confié rentre dans ce cadre là. Il consiste à caractériser puis à classifier certaines formes de battements cardiaques sur des enregistrements électro-cardiographiques (Holter) de 30mn. Pour ce faire, l'outil d'apprentissage retenu est le SVM.

Les *Support Vector Machines* (SVMs) sont une technique d'apprentissage artificiel performante proposées par V.Vapnik. Elle permet d'aborder des problèmes très divers comme la classification et la régression. Le principal objectif des SVMs appliqués à la classification est de construire un hyperplan séparateur optimal entre deux classes, c'est-à-dire, avec la plus grande marge. Lorsqu'une solution linéaire n'est pas possible, la méthode réalise une projection de l'espace d'entrée dans un espace de caractéristiques de dimension plus élevée. Le point fort des SVMs réside dans leur capacité de généralisation et au nombre réduit de paramètres à régler. Ainsi cette méthode est particulièrement bien adaptée au traitement des données de très haute dimension. Les SVM sont appliqués avec une efficacité remarquable à la reconnaissance de caractères manuscrits, au traitement d'images, au diagnostic médical, etc.

Problématique

Le Holter est un examen de battements cardiaques appelé Electrocardiogramme (ECG) de longue durée. Cet examen enregistre environ 100 000 battements cardiaques. Traiter manuellement tous ces battements est une tâche fastidieuse voire impossible, sans compter le temps qu'il faudrait pour le faire.

La caractérisation d'un signal consiste à prélever les caractéristiques du signal avec lesquelles on peut le reconstituer.

Les algorithmes d'apprentissage étant complexes, l'acquis d'algorithme à utiliser nécessite une étude approfondie de celui-ci.

Méthodologie de recherche

Pour étudier notre thème de recherche, nous nous sommes basés d'abord sur l'exploitation des ouvrages relatifs à la classification par SVM d'une manière générale. Nous nous sommes inspirés des mémoires et thèse de recherches ayant porté sur notre sujet et référés aux sites internet pour la compréhension de certains concepts.

Pour répondre à l'ensemble de questions que nous nous sommes posées, nous avons structuré notre travail en quatre chapitres :

Le premier chapitre est consacré à l'apprentissage et l'aide à la décision. Nous présenterons la caractérisation d'une donnée suivie de notions sur l'apprentissage, ces différents objectifs, les différents types d'apprentissages et quelque classifieurs. Parmi lesquels nous relevons le SVM.

Nous exposons dans le second chapitre, le fondement mathématique des SVMs. Nous présenterons l'algorithme détaillé du SVM dans le cas linéairement séparable et non linéairement séparable.

Pour caractériser notre application qui consiste à classifier des battements cardiaques, nous avons consacré le troisième chapitre pour expliquer comment l'extraction des caractéristiques a été faite.

Le dernier chapitre met en évidence l'objectif même de ce mémoire. Les résultats obtenus aux différentes étapes de notre algorithme seront présentés et discutés.

Chapitre I

Apprentissage et aide à la décision

I.1 Introduction

En une vingtaine d'années, l'apprentissage artificiel est devenu une branche majeure des mathématiques appliquées, à l'intersection des statistiques et de l'intelligence artificielle. Son objectif est de réaliser des modèles qui apprennent par des exemples: il s'appuie sur des données numériques (résultats de mesures ou de simulations). L'objectif des chercheurs en intelligence artificielle vise à programmer des machines capables d'effectuer des tâches qui requièrent de l'intelligence. Ces domaines d'applications sont multiples: fouille de données (FD), bioinformatique, génie des procédés, aide au diagnostic médical, télécommunications, interface cerveau-machines, et bien d'autres.

La fouille de données *Data Mining* permet l'extraction Automatique d'informations prédictives à partir de gros volumes de données. Le développement récent de la fouille de données (depuis le début des années 1990) est lié à plusieurs facteurs visant différents domaines, le nombre important de données à traiter et les tâche à effectuer sont pratiquement impossible. Elle représente un mélange d'outils provenant de la Statistique, de l'Intelligence Artificielle et de l'Informatique (reconnaissance de formes)[1].

I.2 Caractérisation de données

Une donnée est une entité caractérisant un objet. Elle est constituée d'attributs. Elle est aussi appelée exemple, échantillon, point, et souvent vecteur. Un ensemble de données est souvent représenté par une matrice dont le nombre de lignes est le nombre d'exemples tandis que le nombre de colonne représente le nombre d'attributs. Un attribut est un descripteur appelé aussi variable, champs, caractéristique ou observation.

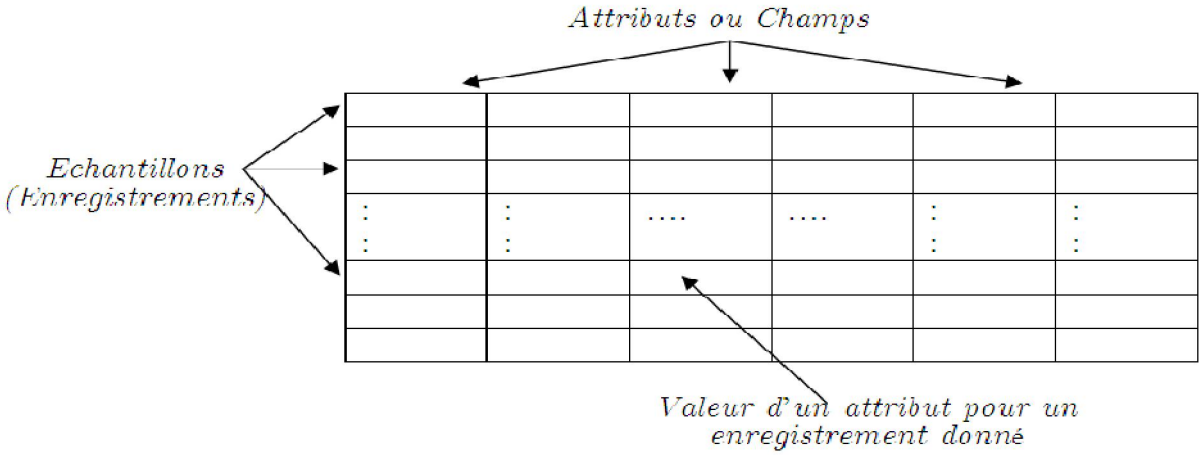


Figure I.1 : Plusieurs données (échantillons) décrites par plusieurs critères (attributs).

I.2.1 Différents types d'attributs

Un attribut peut être de nature qualitative ou quantitative.

- Il est qualitatif si on ne peut pas en faire une moyenne, sa valeur est d'un type défini (ex : masculin, féminin, couleur,...).
- Il est quantitatif s'il peut prendre une valeur entière ou réelle. On peut leur appliquer les opérateurs arithmétiques habituels (ex : poids, la taille, les dimensions d'un objet...).

I.2.2 Natures d'attributs

La valeur d'un attribut peut être nominale ou catégoriel, elle appartient à un ensemble fini et non ordonné (ex : couleur). Elle peut être ordinale (température) ou absolue (nombre). Ces différentes natures entrainent que les opérations qu'on peut faire sur ces attributs ne sont pas les mêmes.

I.2.3 Données bruitées

Dans le cas où une donnée contient une où plusieurs données non valide on dira que la donnée est bruitée. Pour cela un ensemble de prétraitement est nécessaire [2].

I.2.4 Prétraitement des données

Les données doivent être nettoyées puisqu'elles peuvent contenir plusieurs types d'anomalies :

- des données peuvent être omises à cause des erreurs. Dans ce cas, il faut remplacer ces données ou éliminer complètement leurs enregistrements.
- Des données peuvent être incohérentes c'est-à-dire, sortent des intervalles permis. On doit les écarter où les normaliser.

1.2.4.1Normalisation des données

Pour que ces données aient la même influence sur la construction du modèle, leur normalisation est nécessaire car les attributs sont généralement exprimés avec des unités différentes et ont des ordres de grandeur différents. Chaque attribut suivra alors une distribution gaussienne centrées et réduites.

Soit un nuage de points à N échantillons décrits par P attributs. Pour obtenir un nuage de points centré réduit, chaque observation x_i^k va être remplacée par :

$$a_i^k = \frac{x_i^k - \overline{x^k}}{\sigma^k} \qquad \text{(I.1)}$$

Où i=1,…N, k=1,…P, $\overline{x^k}$ est la moyenne des observations x_i^k et σ^k leur écart type.

$$\overline{x^k} = \frac{1}{N}\sum_{i=1}^{N} x_i^k \qquad (I.2)$$

$$\sigma^k = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left|x_i^k - \overline{x^k}\right|^2} \qquad (I.3)$$

Comme la matrice des données a été centrée et réduite, on va travailler à présent sur des variables a_i^k telles que :

$$\overline{a^k} = 0 \; et \; \sigma^k = 1$$

Le prétraitement comporte aussi la réduction des données qui permet de réduire le nombre d'attributs pour accélérer les calculs et représenter les données sous un format optimal pour l'exploration. Une méthode largement utilisée dans ce contexte, est l'analyse en composantes principales (ACP).

1.2.4.2 Analyse en composantes principales

L'ACP consiste à réduire les paramètres de description qui ont une mineur influence et garder les plus discriminants. Elle se traduit par les étapes suivantes :

1. Construction de matrice de covariance : c'est une matrice carrée et symétrique de dimension égale au nombre d'attributs. Elle permet de mesurer la corrélation entre les différents attributs. Cette matrice est donnée par :

$$A = \begin{bmatrix} var(a^1) & cov(a^1a^2) \ldots\ldots\ldots & cov(a^1a^P) \\ cov(a^2a^1) & var(a^2) \ldots\ldots\ldots & cov(a^2a^P) \\ \ldots\ldots\ldots\ldots\ldots\ldots & \ldots\ldots\ldots\ldots\ldots\ldots & \ldots\ldots\ldots\ldots\ldots\ldots \\ cov(a^Pa^1) & cov(a^Pa^2) \ldots\ldots\ldots & var(a^P) \end{bmatrix} \qquad (I.4)$$

Les variables étant centrées et réduites, leurs variances sont toutes égales à 1. De plus A est une matrice symétrique. Alors :

$$A = \begin{bmatrix} 1 & cov(a^1a^2) \ldots\ldots\ldots & cov(a^1a^P) \\ cov(a^2a^1) & 1 \ldots\ldots\ldots & cov(a^2a^P) \\ \ldots\ldots\ldots\ldots\ldots\ldots & \ldots\ldots\ldots\ldots\ldots\ldots & \ldots\ldots\ldots\ldots\ldots\ldots \\ cov(a^Pa^1) & cov(a^Pa^2) \ldots\ldots\ldots & 1 \end{bmatrix} \qquad (I.5)$$

2. Analyse de la matrice des covariances : consiste à calculer les valeurs propres λ_i racines du polynôme caractéristique $det(A - \lambda I) = 0$ puis les vecteurs propres V_i correspondants. Par ailleurs, on sait que la trace de cette matrice est égale à la somme des valeurs propres et égale à la somme des variances.

$$Tr(A) = \sum_{i=1}^{P} \lambda_i = \sum_{i=1}^{P} var(a^i) \text{ =Inertie totale} \quad \text{(I.6)}$$

On remarque alors que la valeur propre la plus élevée est celle qui représente la plus grande inertie.

3. Les composantes principales : consiste à extraire les vecteurs propres les plus discriminants en ordonnant les valeurs propres du plus grand au plus petit (chaque valeur propre possède son vecteur propre) et calculer l'inertie cumulée sachant que l'inertie totale :

$$I_T = \frac{1}{N}\left[\sum_{i=1}^{N}(x_i^1 - \overline{x^1})^2 + \sum_{i=1}^{N}(x_i^2 - \overline{x^2})^2 + \cdots \sum_{i=1}^{N}(x_i^p - \overline{x^p})^2\right] \quad \text{(I.7)}$$

$$I_T = \sum_{k=1}^{P} var(a^k) \quad \text{(I.8)}$$

$$I_{c1} = \frac{\lambda_1}{I_T} ; I_{c2} = \frac{\lambda_1+\lambda_2}{I_T} ; \ldots I_{cp} = \frac{\lambda_1+\lambda_2+\ldots\lambda_p}{I_T} = 1$$

Ce calcul nous aide à approcher la description du phénomène toute en réduisant le nombre d'attribut. Le choix du nombre d'attributs est traduit par un pourcentage de ressemblance. Une autre méthode graphique permet le choix des composantes principales appelé éboulis, son principe consiste à rechercher, s'il existe, un coude dans le graphe et de ne conserver que les valeurs propres jusqu'à ce coude.

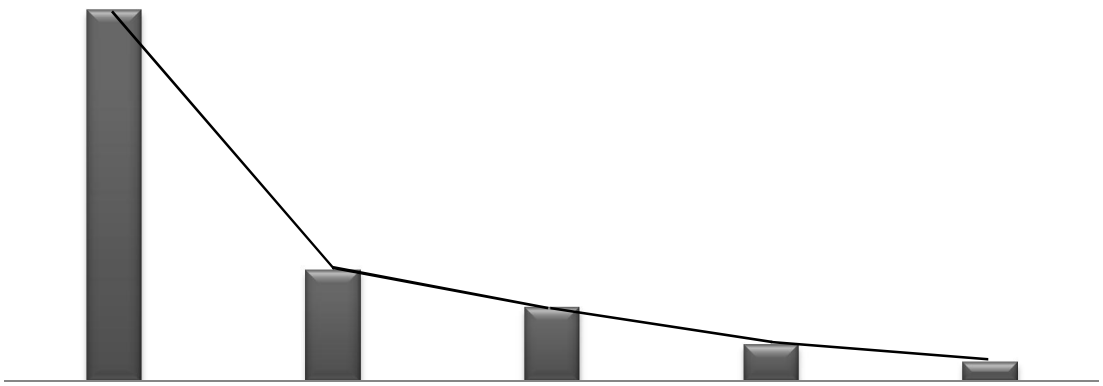


Figure I.2 : Éboulis des valeurs propres

Il ne reste plus qu'à projeter nos données sur les axes artificiels qui sont le produit scalaire des exemples normalisés et les vecteurs propres choisis.

$$\begin{pmatrix} a_1^1 a_1^2 \dots . a_1^p \\ a_2^1 a_2^2 \dots . a_1^p \\ \dots \dots \dots \dots \dots \\ a_n^1 a_n^2 \dots . a_n^p \end{pmatrix} . \begin{pmatrix} v_1^1 \\ v_2^1 \\ \dots \\ v_p^1 \end{pmatrix} . \begin{pmatrix} v_1^2 \\ v_2^2 \\ \dots \\ v_p^2 \end{pmatrix} \dots . \begin{pmatrix} v_1^p \\ v_2^p \\ \dots \\ v_p^p \end{pmatrix} = \begin{pmatrix} C_1^1 C_1^2 \dots C_1^p \\ C_1^1 C_1^2 \dots C_1^p \\ \dots \dots \dots \dots \\ C_n^1 C_n^2 \dots . C_n^p \end{pmatrix} \quad \text{(I.9)}$$

On appel axes principaux d'inertie, les axes dont la direction est donnée par les vecteurs propres de A. Il y'en a P.

Si quelques valeurs propres ont des valeurs bien plus importantes que les autres, cela signifie que l'essentiel des informations est donné par les axes principaux correspondants.

Le premier axe est celui associé à la plus grande valeur propre. On le note C^1

Le deuxième axe est celui associé à la deuxième valeur propre. On le note C^2 etc...

La situation idéale est bien entendu, lorsque une ou deux valeurs propres sont très importantes par rapport aux autres comme la méthode graphique représentée à la figure I.2 ; les axes principaux sont alors C^1 et C^2.

$$C = \begin{pmatrix} C_1^1 C_1^2 \\ C_1^1 C_1^2 \\ \dots \dots \\ C_n^1 C_n^2 \end{pmatrix} \quad \text{(I.10)}$$

I.3 Notions d'apprentissage statistique

L'apprentissage statistique, appelé aussi *machine learning* (traduit en français par apprentissage numérique, apprentissage automatique ou encore apprentissage artificiel) est un ensemble de méthodes et d'algorithmes qui permettent à un modèle d'apprendre un comportement grâce à des exemples.

Pour mettre en œuvre un projet d'apprentissage et évaluer ces performances, plusieurs étapes sont nécessaires :

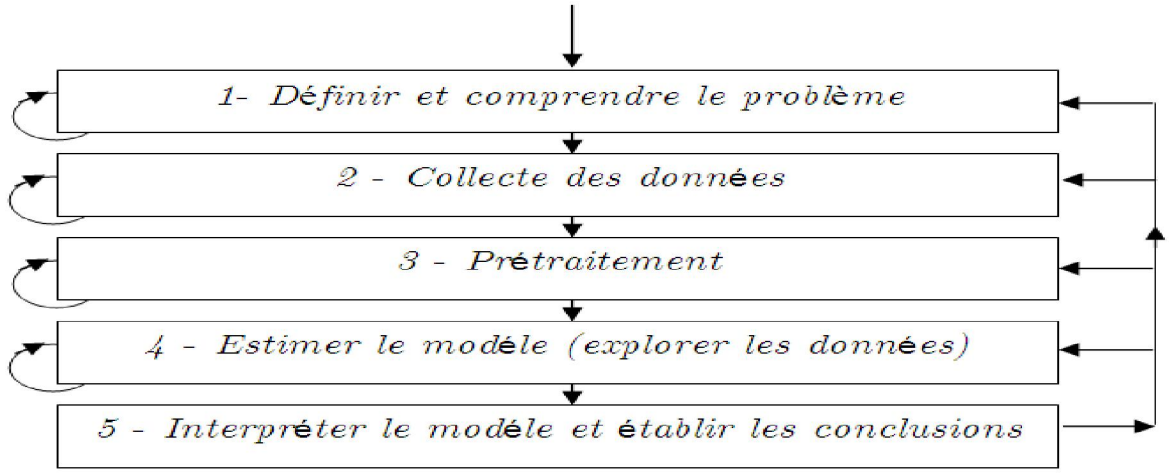


Figure I.3 : Processus d'apprentissage artificiel.

Définition et compréhension du problème : il est indispensable de comprendre la signification des données et le domaine à explorer. En effet, Avec la compréhension du problème, on peut préparer les données nécessaires à l'exploration et interpréter correctement les résultats obtenus.

Collecte des données : dans cette étape, on s'intéresse à la manière dont les données sont générées et collectées. Ces données n'ont pas toujours le même format et la même structure.

Prétraitement : comme expliqué précédemment, il consiste à nettoyer les données des différentes anomalies.

Estimation du modèle : Dans cette étape, on doit choisir la bonne technique pour l'exploration des données. Des techniques telles que les réseaux de neurones, les arbres de décision, les réseaux bayésiens, le clustering, ... sont utilisées. Généralement, l'implémentation se base sur plusieurs de ces techniques, puis on choisit le bon résultat.

Interprétation du modèle et établissement des conclusions : généralement, l'objectif est d'aider à la prise de décision en fournissant des modèles compréhensibles aux utilisateurs. Les expériences montrent que les modèles simples sont plus compréhensibles mais moins précis, alors que ceux complexes sont plus précis mais difficiles à interpréter.[3]

I.3.1 Objectifs de l'apprentissage

Dans ce domaine, plusieurs objectifs peuvent être poursuivis :

Classification

La classification consiste à prédire la classe à laquelle appartient un échantillon de données par approche probabiliste, par approximation, par construction d'un modèle arborescent ou par apprentissage automatique.

L'analyse des clusters

Le *clustering* (ou la segmentation) est le regroupement d'enregistrements ou des observations en classes d'objets similaires.

La différence entre le *clustering* et la classification est que dans le *clustering* il n'y a pas de variables sortantes. La tâche de *clustering* ne classifie pas, n'estime pas, ne prévoit pas la valeur d'une variable sortantes. Au lieu de cela, les algorithmes de *clustering* visent à segmenter la totalité de données en des sous groupes relativement homogènes. Ils maximisent l'homogénéité à l'intérieur de chaque groupe et la minimisent entre les différents groupes.

L'estimation

L'estimation est similaire à la classification à part que la variable de sortie est numérique plutôt que catégorique. En fonction des autres champs de l'enregistrement, l'estimation consiste à compléter une valeur manquante dans un champ particulier. Par exemple on cherche à estimer la lecture de la tension systolique d'un patient dans un hôpital, en se basant sur l'âge du patient, son genre, son indice de masse corporelle et le niveau de sodium dans son sang. La relation entre la tension systolique et les autres données vont fournir un modèle d'estimation. Par la suite nous pouvons appliquer ce modèle à d'autres cas.[3]

La régression

Dans le cas de la régression, l'entrée n'est pas associée à une classe mais à une ou plusieurs quantités continues. Ainsi, l'entrée pourrait être les attributs d'un objet et l'étiquette son revenu [4].

I.3.2 Différents types d'apprentissage

a. Apprentissage supervisé

Dans l'apprentissage supervisé les données fournies à l'entrée sont étiquetées ; les exemples fournis sont sous la forme de couples entrée sortie (*xi, yi*). L'objectif est d'inférer la sortie *y* pour une nouvelle entrée *x*.

On parle de classification si $yi \in \{-1,1\}$ plus généralement $yi \in N$. Et on parle de régression si $yi \in R$.

b. Apprentissage non supervisé

L'apprentissage non supervisé consiste à apprendre à classer sans supervision ; les exemples fournis ne sont que des entrées *xi*. L'objectif est alors de résumer l'espace des *xi* possibles.

Il existe plusieurs types d'algorithmes d'apprentissage non supervisé tels que les algorithmes de partitionnements et les algorithmes de classification hiérarchique.

c. Apprentissage semi supervisé

Dans l'apprentissage semi supervisé, certaines données sont étiquetées et d'autres ne le sont pas. Il réalise les mêmes tâches que celles réalisées en apprentissage supervisé, à la différence qu'il fait usage des données non étiquetées.

I.3.3 Méthode d'apprentissage supervisé

Dans l'apprentissage supervisé, un expert (ou oracle) doit préalablement correctement étiqueter des exemples. L'apprenant peut alors trouver ou approximer la fonction qui permet d'affecter la bonne « étiquette » à ces exemples.

a. Le modèle générale de l'apprentissage supervisé

Il se décompose de trois parties

Un environnement : il engendre des entrées x_i tirées indépendamment et identiquement distribuées.

Un superviseur (oracle) : retourne pour chaque entré x_i une étiquette $u_i = f(x_i)$.

Un apprenant : capable de réaliser une fonction h à partir d'un espace d'hypothèses H qui prédit au mieux la réponse du superviseur $y_i = h(x_i)$.

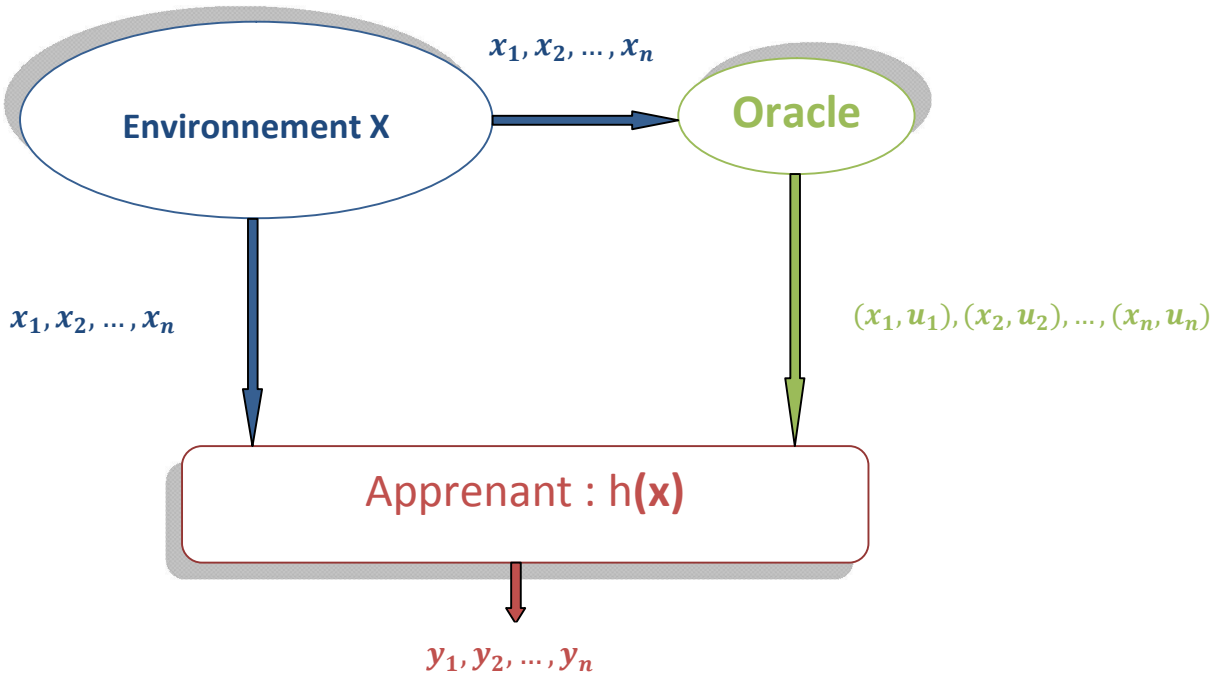


Figure I.4 : Modèle d'un apprentissage supervisé

b. La tâche d'apprentissage

Consiste à chercher dans un espace d'hypothèse H une fonction h qui approxime au mieux la réponse désirable du superviseur. Ce qui implique un problème d'approximation de la fonction f.

Pour chaque entrée x_i, une perte $L(h(x_i), u_i)$ qui évalue le coût d'avoir pris la décision $y_i = h(x_i)$ quand la réponse désirée est $u_i = f(x_i)$. La forme de cette perte dépond principalement de la tâche à accomplir :

$$Cas\ d'une\ classification : \quad L(h(x_i), u_i) = \begin{cases} 0 & si \quad u_i = h(x_i) \\ 1 & si \quad u_i \neq h(x_i) \end{cases} \tag{I.11}$$

$$Cas\ d'une\ régression : \quad L(h(x_i), u_i) = [h(x_i) - u_i]^2 \tag{I.12}$$

Nous définissons ainsi le risque réel par l'espérance de perte ou le coût :

$$R_{reel}(h) = \int L(h(x_i), u_i)\, dP(x, u) \tag{I.13}$$

Le risque réel ne peut pas être directement minimisé vu que la distribution de probabilité $P(x, u)$ est inconnue. Pour ce faire, on approxime le minimum du risque réel par le minimum du risque empirique. On appelle cette mesure un risque empirique car elle est

mesurée empiriquement sur les données d'apprentissage. Ce risque est la moyenne des coûts mesurés pour chaque exemple d'apprentissage. Il est sous la forme :

$$Remp(h) = \frac{1}{N} \sum_{i=1}^{N} L(h(x_i), u_i) \quad \text{(I.14)}$$

Le risque empirique Remp converge vers le risque réel R_{reel} lorsque le nombre de vecteurs d'apprentissage N tend vers l'infini.[4]

I.4 Les différents types de Classifieurs

Il existe différents types de classifieurs parmi lesquels on cite:

I.4.1 Les k plus proches voisins

Les k plus proches voisins *K-nearest neighbor (K-NN)* est une méthode d'apprentissage supervisée qui résonnent avec le principe sous-jacent. Elle diffère des autres méthodes d'apprentissages par l'absence du modèle induit par des exemples. Les données restent telles quelles, elles sont simplement stockées en mémoire. Selon le nombre k choisi, un nouvel exemple sera classé dans la classe majoritaire pour les k voisins sélectionnés.

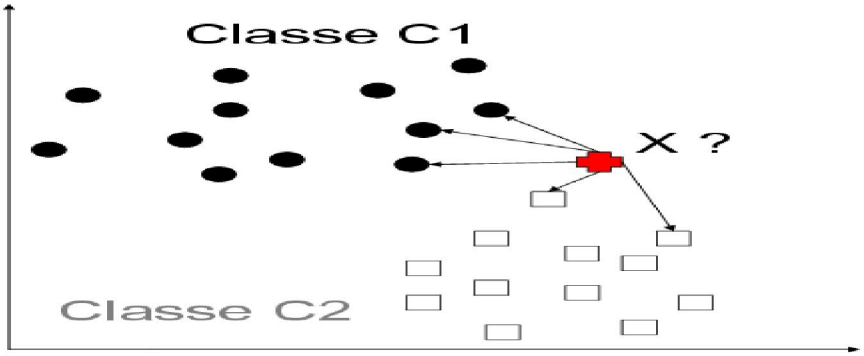


Figure I.5 : Les plus proches voisins.

Avantages

- Absence d'apprentissage : Ce sont les échantillons pris en considération, qui constituent le modèle.
- Clarté des résultats : bien que la méthode ne produise pas de règle explicite, la classe attribuée à un exemple peut être expliquée en exposant les plus proches voisins qui ont imposé cette attribution.

- Données hétérogènes : la méthode peut s'appliquer dès qu'il est possible de définir une distance sur les champs.
- Grand nombre d'attributs : la méthode permet de traiter des problèmes avec un grand nombre d'attributs. Cependant, plus le nombre d'attributs est important, plus le nombre d'exemples doit être grand.

Inconvénients

- Sélection des attributs pertinents : pour que la notion de proximité soit pertinente, il faut que les exemples couvrent bien l'espace et soient suffisamment proches les uns des autres. Si le nombre d'attributs pertinents est faible relativement au nombre total d'attributs, la méthode donnera de mauvais résultats.
- Le temps de classification : si la méthode ne nécessite pas d'apprentissage, tous les calculs doivent être effectués lors de la classification.
- Définir les distances et nombres de voisins : les performances de la méthode dépendent du choix de la distance, du nombre de voisins et du mode de combinaison des réponses des voisins. Le calcul des distances euclidiennes impose que les attributs constituent un espace orthonormé.

I.4.2 Réseaux de neurones

Les réseaux de neurones forment une classe de classifieurs supervisés. Ils sont inspirés de la structure neurophysiologique des neurones. Un neurone formel est l'unité élémentaire d'un système modélisé d'un réseau de neurone. A la réception de signaux provenant d'autres neurones du réseau, un neurone formel réagit en produisant un signal de sortie qui sera transmis à d'autres neurones du réseau. Le signal reçu à la sortie d'un neurone est une combinaison linéaire des sorties et neurones précédents. Le signal de sortie est une fonction de cette somme pondérée :

$$y_j = f\left(\sum_{i=1}^{R} w_{ij}.x_i\right) \qquad \text{(I.15)}$$

Avec y_j la sortie du neurone formelle j, x_i les signaux reçus par le neurone j de la part des neurones i, w_{ij} les poids des interconnexions entre les neurones i et j. Selon l'application, la fonction f, appelée fonction d'activation, est le plus souvent une fonction identité, sigmoïde, tangente hyperbolique ou une fonction linéaire par morceaux.

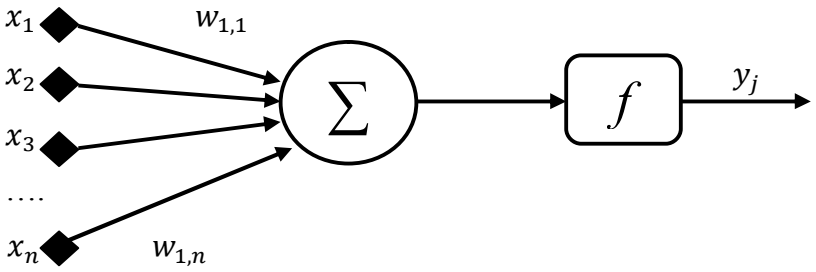


Figure I.6 : Neurone formel.

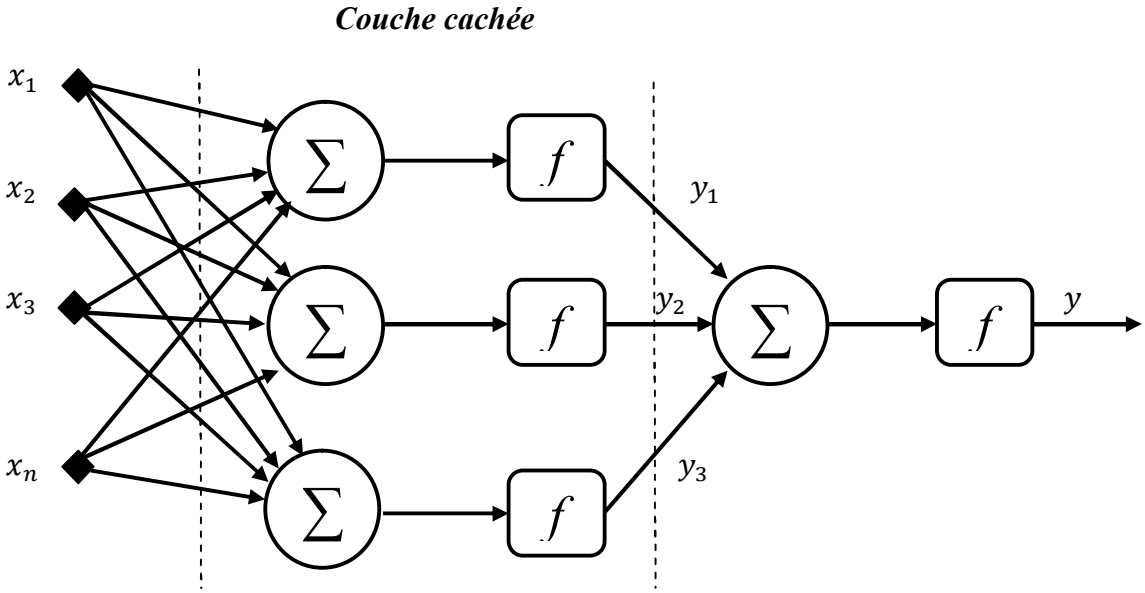


Figure I.7 : Perceptron multicouche avec une couche cachée et une couche de sortie.

Avantages

- Lisibilité du résultat : le résultat de l'apprentissage est un réseau constitué de cellules organisées selon une architecture, définies par une fonction d'activation et un très grand nombre de poids à valeurs réelles.
- Les données réelles : les réseaux traitent facilement les données réelles "préalablement normalisées" et les algorithmes sont robustes au bruit.
- Classification efficace : le calcul d'une sortie à partir d'un vecteur d'entrée est un calcul très rapide.
- Leur utilisation est diverse. En plus des tâches de classification, les RNN peuvent être utilisés en filtrage, en modélisation…etc.

Inconvénients

- Détermination de l'architecture du réseau est complexe.

- Paramètres difficiles à interpréter (boite noire).
- Difficulté de paramétrage surtout pour le nombre de neurone dans la couche cachée.
- Temps d'apprentissage: l'échantillon nécessaire à l'apprentissage doit être suffisamment grand et représentatif des sorties attendues.

I.4.3 Arbres de décision

Les arbres de décision sont les plus populaires des méthodes d'apprentissage. Comme toute méthode d'apprentissage supervisé, les arbres de décision utilisent des exemples. Si l'on doit classer des exemples dans des catégories, il faut construire un arbre de décision par catégorie. Pour déterminer à quelle catégorie appartient un nouvel exemple, on utilise l'arbre de décision de chaque catégorie auquel on soumet le nouvel exemple à classer.

Chaque arbre répond Oui ou Non (il prend une décision).Concrètement, chaque nœud d'un arbre de décision contient un test (un IF...THEN) et les feuilles ont les valeurs Oui ou Non. Chaque test regarde la valeur d'un attribut de chaque exemple. Pour construire l'arbre de décision, il faut trouver quel attribut tester à chaque nœud.

C'est un processus récursif. Pour déterminer quel attribut tester à chaque étape, on utilise un calcul statistique qui détermine dans quelle mesure cet attribut sépare bien les exemples Oui/Non. On crée alors un nœud contenant ce test, et on crée autant de descendants que de valeurs possibles pour ce test.

L'arbre de décision classe trop bien les exemples, mais est mauvais pour généraliser, c'est-à-dire qu'il prédit mal la classification (Oui /Non) de nouvelles instances.

Soit A, B et C trois entiers, pour les ordonnées par un arbre de décision, un ensemble de test doit être réalisé, la figure I.8 illustre cet exemple :

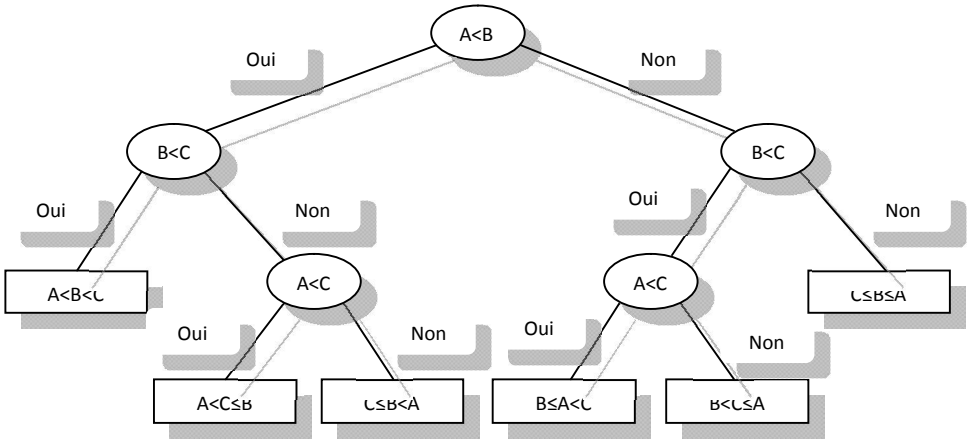


Figure I.8 : Ordonnancement de trois entiers par Arbre de décision.

Avantages

- Adaptabilité aux attributs de valeurs manquantes : les algorithmes peuvent traiter les valeurs manquantes (descriptions contenant des champs non renseignés) pour l'apprentissage, mais aussi pour la classification.
- Bonne lisibilité du résultat : un arbre de décision est facile à interpréter et à la représentation graphique d'un ensemble de règles. Si la taille de l'arbre est importante, il est difficile d'appréhender l'arbre dans sa globalité. Cependant, les outils actuels permettent une navigation aisée dans l'arbre (parcourir une branche, développer un nœud, élaguer une branche) et, le plus important, est certainement de pouvoir expliquer comment est classé un exemple par l'arbre, ce qui peut être fait en montrant le chemin de la racine à la feuille pour l'exemple courant.
- Traitement de tout type de données : l'algorithme peut prendre en compte tous les types d'attributs et les valeurs manquantes. Il est robuste au bruit.
- Sélectionne des variables pertinentes : l'arbre contient les attributs utiles pour la classification. L'algorithme peut donc être utilisé comme prétraitement qui permet de sélectionner l'ensemble des variables pertinentes pour ensuite appliquer une autre méthode.
- Donne une classification efficace : l'attribution d'une classe à un exemple à l'aide d'un arbre de décision est un processus très efficace (parcours d'un chemin dans un arbre).
- Disponibilité des outils : les algorithmes de génération d'arbres de décision sont disponibles dans tous les environnements de fouille de données.
- Méthode extensible et modifiable : la méthode peut être adaptée pour résoudre des tâches d'estimation et de prédiction. Des améliorations des performances des algorithmes de

base sont possibles grâce aux techniques qui génèrent un ensemble d'arbres votant pour attribuer la classe.

Inconvénients

- Méthode sensible au nombre de classes : les performances tendent à se dégrader lorsque le nombre de classes devient trop important.
- Manque d'évolutivité dans le temps : l'algorithme n'est pas incrémental, c'est-à-dire, que si les données évoluent avec le temps, il est nécessaire de relancer une phase d'apprentissage sur l'échantillon complet (anciens exemples et nouveaux exemples).[5][6]

I.4.4 Support Vector Machines

Les *Support Vector Machines* constituent une technique d'apprentissage supervisée introduite en fin des années 90. Grâce à son fondement mathématique et à ses performances, cette technique a ouvert un domaine de recherche très actif et un grand éventail d'applications.

Le SVM consiste à chercher le meilleur hyperplan qui sépare linéairement deux classes tout en les repoussant au maximum. Lors de sa phase d'apprentissage, le SVM vise à maximiser la marge entre les deux classes d'apprentissage. Ce qui lui procure un grand pouvoir de généralisation pendant la phase de test. Cette méthode sera détaillée au chapitre suivant.

Avantages

- Les SVM possèdent des fondements mathématiques solides.
- Les exemples de test sont comparés juste avec les supports vecteur et non pas avec tout les exemples d'apprentissage.
- Décision rapide. La classification d'un nouvel exemple consiste à voir le signe de la fonction de décision f(x).

Inconvénients

- Classification binaire d'où la nécessité d'utiliser l'approche un-contre-un pour construire un classifieur multiclasse.
- Grande quantité d'exemples en entrées implique un calcul matriciel important.
- Temps de calcul élevé lors d'une régularisation des paramètres de la fonction noyau.

I.5 Conclusion

Ce chapitre donne un aperçu sur la fouille de donnée, un domaine à la croisée de plusieurs disciplines. Dans ce domaine, Les objectifs et les applications sont divers. Le volet sur lequel nous nous focalisons dans ce travail est la classification vue que la problématique posée est la reconnaissance automatique d'extrasystoles ventriculaires.

Après avoir exposé les différents types de classifieurs, nous avons choisi la classification par SVM car il possède un pouvoir de généralisation et un nombre réduit de paramètres à régler sans parler des bonnes performances qu'il présente par rapport aux autres méthodes. Dans le chapitre suivant, nous allons aborder le fondement mathématique des SVMs et ainsi mettre en évidence leur principe et leurs performances.

Chapitre II

Support Vector Machines

II.1 Introduction

Les "*Support Vector Machines*", ou Séparateurs à Vaste Marge (SVM) sont un ensemble de techniques d'apprentissage supervisée destinés à résoudre les problèmes de discrimination et de régression. Initiée par Vladimir Vapnik comme méthode de classification binaire qui cherche le meilleur hyperplan qui sépare linéairement les exemples positifs des exemples négatifs en garantissant que la marge entre le plus proche des positifs et des négatifs soit maximal. Ils sont ensuite développés par l'introduction de fonction dites noyau (kernel) par Boser afin de régler les problèmes de la non linéarité, quant à Cortes & al pour traiter les cas de données non linéairement séparables proposent une version régularisée des SVM qui tolère les erreurs d'apprentissage tout en les pénalisant.[7]

II.2 Notions de bases

II.2.1 Hyperplan

Quand on est dans un espace de représentation euclidien, on peut librement faire des hypothèses sur la géométrie des classes ou sur celles de leurs surfaces séparatrices ; ceci permet de mettre au point des techniques d'apprentissage non statistiquement fondées a priori, mais peut être plus faciles à appliquer. La plus simple d'entre elles est de supposer que deux classes peuvent être séparées par une certaine surface (voir figure II.1). Les paramètres qui régissent son équation sont alors les variables à apprendre. Le nombre de paramètres à calculer est minimal si l'on suppose cette surface linéaire. Dans R^p; une surface linéaire est un hyperplan *H;* défini par l'équation : $w^T x + b = 0$; Si deux classes *C*1 *et C*2 sont séparables par *H*, alors tous les points de la première classe sont par exemple tels que : $x \in C_1 \Rightarrow w^T x + b \geq 0$ et de la seconde vérifient alors : $x \in C_2 \Rightarrow w^T x + b < 0$.

On parle d'hyperplan optimal lorsque celui-ci sépare les deux classes en garantissant un maximum d'espace entre les vecteurs de support.

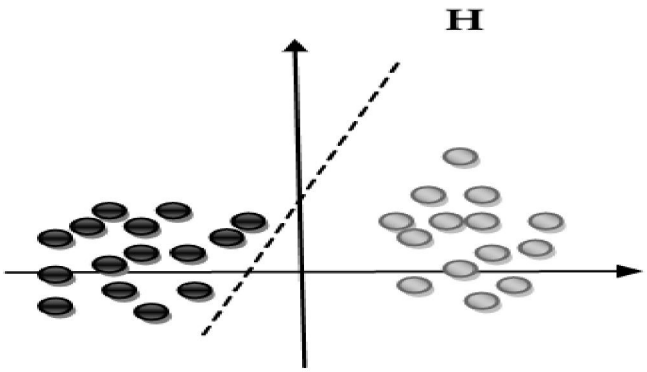


Figure II.1 : Hyperplan séparant deux classes

II.2.2 Support vectors (vecteurs de support)

Ce sont les points les plus proches de l'hyperplan optimal et qui déterminent la marge.

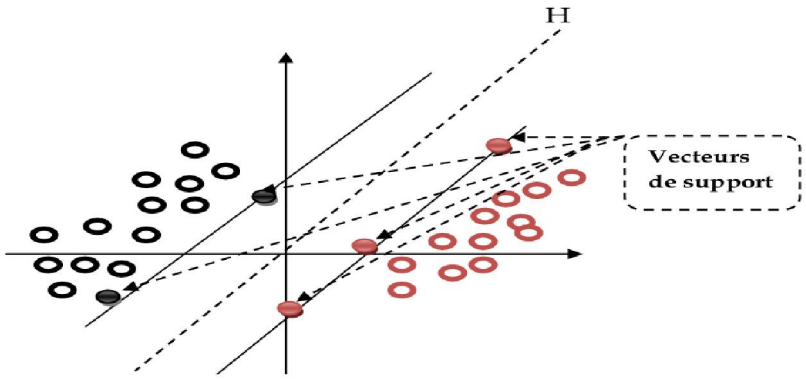


Figure II.2 : Les vecteurs de support

II.2.3 Marge

La marge est la distance euclidienne entre deux vecteurs de support de deux classes différentes.

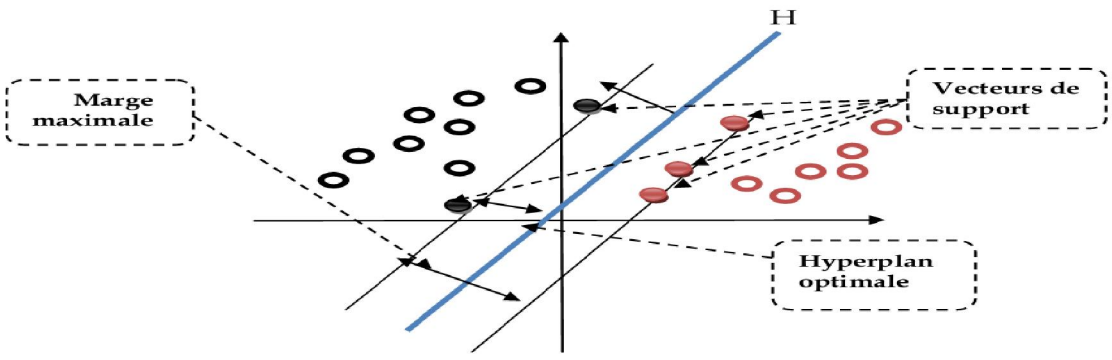


Figure II.3 : Représentation de la marge

II.2.4 Théorie d'apprentissage de Vapnik-Chervonenkis

Afin de résoudre le problème d'apprentissage qui est basé sur la minimisation du risque réel, et vu l'insuffisance de la MRE, Vladimir Vapnik et Chervonenkis [8] ont développé cette théorie où ils ont montrer que

- La condition nécessaire et suffisante pour la consistance de principe MRE est que h soit finie.
- Si F possède une dimension VC finie h, que m>h, avec une probabilité d'erreur au moins égale à 1-η, l'inégalité suivante sera vérifiée :

$$Rréel \leq Remp + \sqrt{\frac{h\left[ln\left(\frac{2m}{h}\right)+1\right]-ln\left(\frac{\eta}{4}\right)}{m}} \quad \text{(II.1)}$$

Le membre droit de l'inégalité appelé le risque garanti est composé de deux termes :

Le risque empirique et une quantité qui dépond du rapport $\frac{m}{h}$ appelée intervalle de confiance puisqu'il représente la différence entre le risque empirique R_{emp} et le risque Rréel.[4]

L'insuffisance de la minimisation du risque empirique **MRE** a incité **VC** de proposer un nouveau principe d'induction « Minimisation du risque structurel » qui a pour but la minimisation du risque réel tout en minimisant conjointement le Remp et l'intervalle de confiance ou la classe de fonction ;

D'une part, en limitant fortement la taille de l'ensemble **H**, on tend à expliquer la relation entre les objets et leurs classes, on parle d'une sur généralisation. Dans ce cas le risque empirique sera élevé mais le modèle ne collera pas aux exemples de trop près.

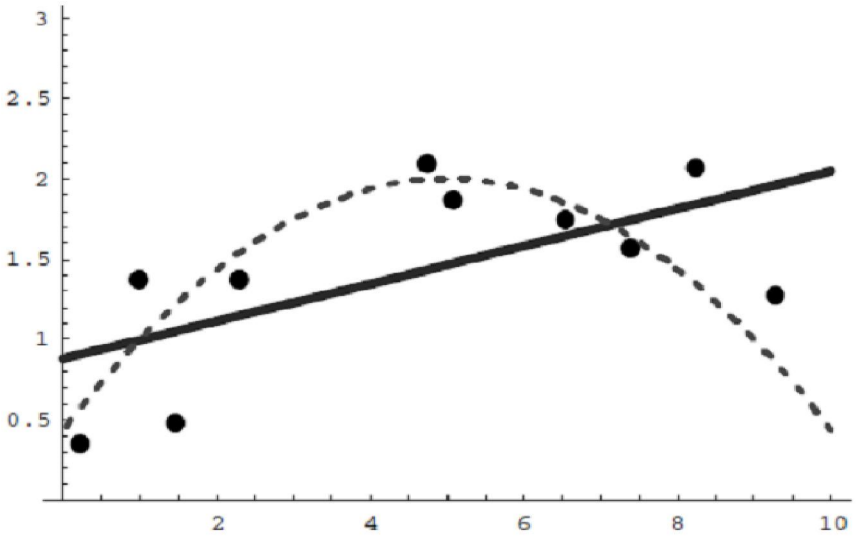


Figure II.4 : Sous apprentissage

D'autre part, si on admet un grand nombre de fonctions, la relation sera modélisée de manière complexe et le bruit associé aux mesures risque également d'être appris. On parle souvent d'apprentissage par cœur parce que le classifieur aura un risque empirique très faible mais ses performances sur d'autres jeux de données seront mauvaises.[5]

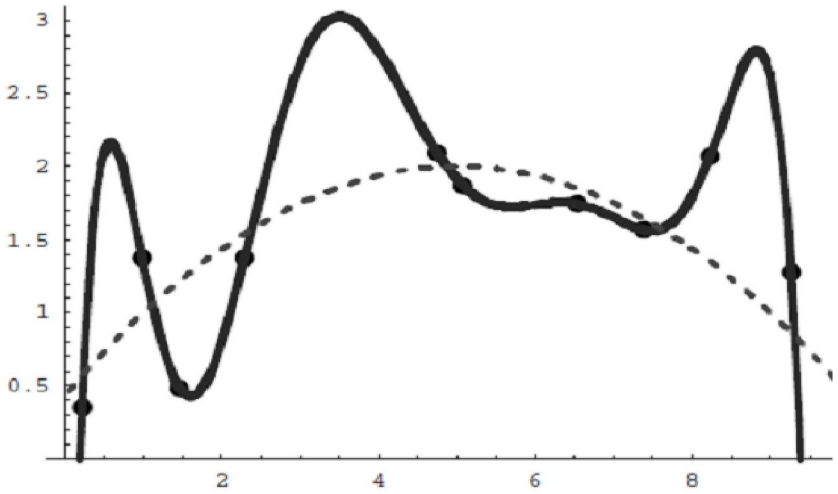


Figure II.5 : Apprentissage par cœur

La restriction des fonctions implémentables nous confronte à un dilemme que les statisticiens appellent biais-variance[9].

La méthode de minimisation de risque structurel **MRS** cherche alors un compromis entre la taille de classe de fonction qui réalise l'approximation et l'approximation sur l'échantillon. L'interaction de la courbe de confiance (variance) et du risque empirique (biais) qui nous donnera la minimisation du risque réel.

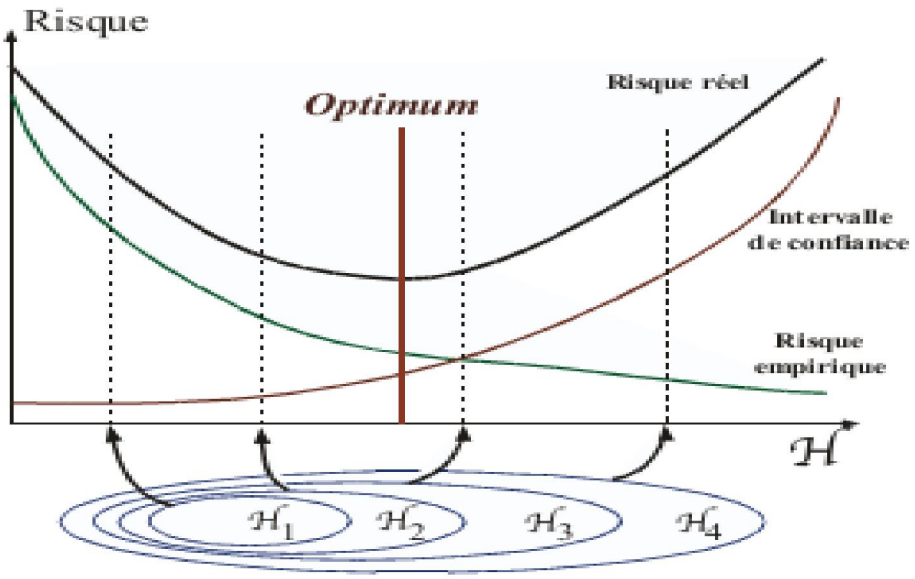


Figure II.6 : Comportement du risque empirique, l'intervalle de confiance et le risque garanti en fonction de la VC dimension

II.3 Principe d'un SVM

Cette technique est une méthode de classification à deux classes. Son objectif est non seulement de séparer les exemples positifs des exemples négatifs mais aussi de repousser au maximum les uns des autres. La méthode cherche alors l'hyperplan qui sépare les deux classes d'exemples, en garantissant que la marge entre les positifs et les négatifs les plus proches de l'hyperplan soit maximale. Cela garantit une meilleure généralisation du modèle car de nouveaux exemples pourraient ne pas être trop similaires à ceux utilisés pour l'apprentissage. L'intérêt de cette méthode est la sélection de vecteurs supports qui représentent les vecteurs discriminant grâce auxquels est déterminé l'hyperplan. Seuls ses vecteurs interviennent dans la solution, ce qui rend le problème moins complexe.

II.3.1 Principes fondamentaux

Maximisation de la marge

Intuitivement, le fait d'avoir une marge plus large procure plus de sécurité lorsqu'on classe un nouvel exemple. De plus, si l'on trouve le classificateur qui se comporte le mieux vis-à-vis des données d'apprentissage, il est clair qu'il sera aussi celui qui permettra au mieux de classer les nouveaux exemples.

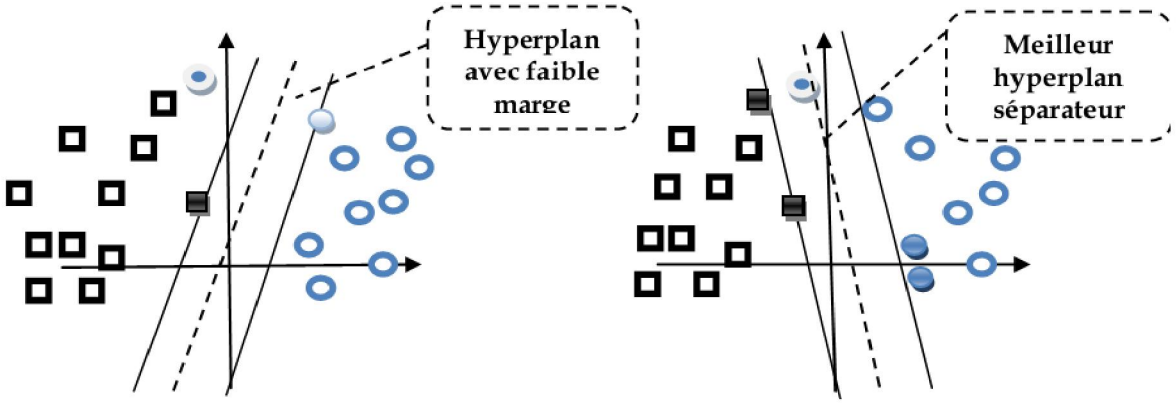


Figure II.7 : Meilleur hyperplan séparateur

Cas linéairement séparable

Les cas linéairement séparables sont les plus simples des SVM car ils permettent de trouver facilement le classificateur linéaire.

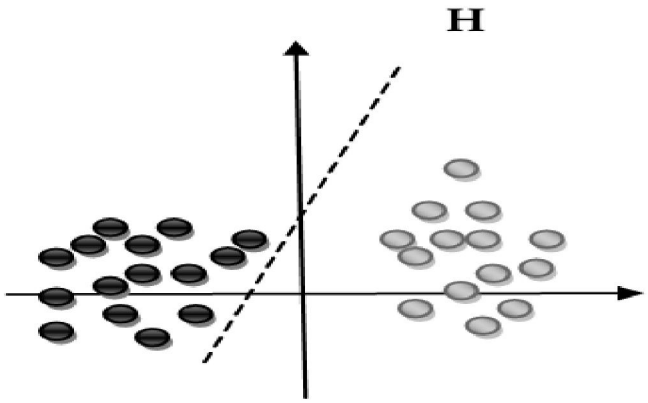


Figure II.8 : Exemple d'un cas linéairement séparable

Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, le classificateur de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables.

Cas non linéairement séparable

Pour surmonter les inconvénients des cas non linéairement séparable, l'idée introduite par V.Vapnick et qui fait d'ailleurs le point fort des SVMs est de projeter l'espace d'entrée sur un espace de plus grande dimension où les données deviennent linéairement séparables. Ce nouvel espace est appelé « espace de redescription ». Intuitivement, plus la dimension de l'espace de redescription est grande, plus la probabilité de pouvoir trouver un hyperplan séparateur entre les exemples est élevée.

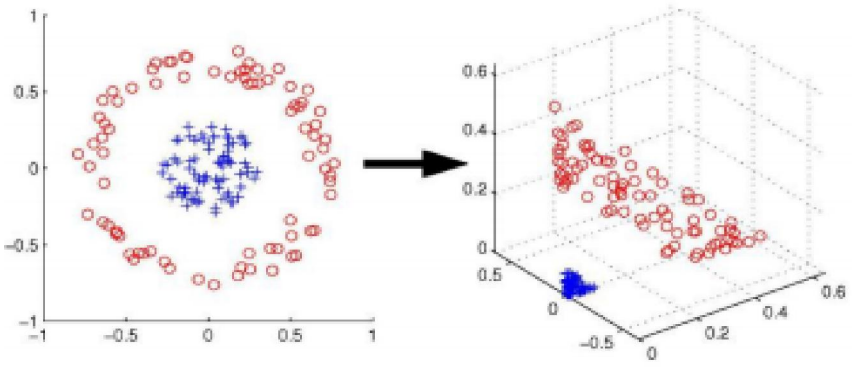


Figure II.9 : Exemple de projection dans un espace de redescription

La figure II.9 montre un exemple de transformation d'un problème de séparation non linéaire dans l'espace de représentation (2 dimensions) en un problème de séparation linéaire dans un espace de redescription de plus grande dimension (3 dimensions). Cette transformation non linéaire est réalisée *via* une fonction noyau.

En pratique, quelques familles de fonctions noyau paramétrables sont connues et il revient à l'utilisateur du SVM d'effectuer des tests pour déterminer celle qui convient le mieux pour son application. On peut citer les exemples de noyaux suivants : polynomial, gaussien, sigmoïde et Laplacien.

II.3.2 Fondement mathématique

II.3.2.1 Cas linéairement séparable

Un classifieur est dit linéaire lorsqu'il est possible d'exprimer sa fonction de décision par une fonction linéaire en x. On peut exprimer une telle fonction par:

$$h(x_i) = < w, x_i > + b = \sum_{j=1}^{p} w_j . x_i^j + b \qquad \text{(II.2)}$$

Où $\boldsymbol{w}$ est le vecteur de poids et $\boldsymbol{b}$ le biais, alors que $\boldsymbol{x}$ est la variable. X est l'espace d'entrée qui correspond à $\boldsymbol{R}^{P}$, où p est le nombre d'attributs des vecteurs d'entrée. P est également la dimension de l'espace d'entée. Notons que l'opérateur $< >$ désigne le produit scalaire usuel.

Pour décider à quelle classe un exemple appartient, il suffit de prendre le signe de la fonction de décision :

$$\begin{cases} \langle w.xi\rangle + b \geq 1 \;\; si \;\; yi = 1 \\ \langle w.xi\rangle + b \;\leq -1 \;\; si \;\; yi = -1 \end{cases}$$

Ce qui est équivalent à :

$$yi(\langle w.xi\rangle + b) \geq 1 \qquad \textbf{\textit{avec i=1…n}} \tag{II.3}$$

Trouver l'hyperplan optimal revient à maximiser la marge M et donc à maximiser la somme des distances euclidienne (d) des deux classes par rapport à l'hyperplan. Ainsi, la marge est donnée par l'expression suivante :

$x_b = x_a - d\frac{w}{\|w\|}$ avec $x_b,\ x_a,\ w$ sont des vecteurs et $\frac{w}{\|w\|}$ vecteur unitaire b étant un point de l'hyperplan $\boldsymbol{h}$, alors il satisfait l'équation

$$\langle w.x_b\rangle + b = 0 \tag{II.4}$$

Dans ce cas, pour xa qui est un point qui n'appartiens pas à l'hyperplan on aura donc :

$$\langle w(x_a - d\frac{\|w\|}{\|w\|})\rangle + b = 0 \tag{II.5}$$

$$\langle w.x_a\rangle - d\frac{\|w.w\|}{\|w\|} + b = 0 \tag{II.6}$$

$$\langle w.x_a\rangle - d\|w\| + b = 0 \tag{II.7}$$

$$d = \frac{(\langle w.x_a\rangle + b}{\|w\|} \tag{II.8}$$

$$|\langle w.x_a\rangle + b| = 1 \tag{II.9}$$

$$d = \frac{1}{\|w\|} \tag{II.10}$$

$$M = 2.d = \frac{2}{\|w\|} \tag{II.11}$$

Le problème devient :

$$PQ1\begin{cases} min\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\boldsymbol{\varepsilon_i} \\ Sc \quad y_i[\langle w.x_i\rangle + b] \geq 1 - \boldsymbol{\varepsilon_i} \end{cases} \qquad (II.12)$$

$$Avec\ \ i = 1 \ldots n.$$

a. *Marge dure* ($\boldsymbol{tous\ les\ \varepsilon_i = 0\ et\ C = 0\ dans\ (PQ1)}$)

Il s'agit d'un problème quadratique convexe sous contraintes linéaires de forme primal dont la fonction objective est à minimiser. Cette fonction objective est le carré de l'inverse de la double marge. L'unique contrainte stipule que les exemples doivent être bien classés et qu'ils ne dépassent pas les hyperplans canoniques.

Dans cette formulation, les variables à fixer sont les composantes *w* et *b*. Le vecteur $\boldsymbol{w}$ possède un nombre de composantes égal à la dimension de l'espace d'entrée. Généralement dans ce type de cas, on résout la forme duale du problème. Le passage du problème primal au dual introduit trois principes mathématiques qui sont : principe de Fermat, principe de Lagrange et principe de Kuhn-Tucker.

Nous devons faire rentrer les contraintes dans la fonction objective et de pondérer chacune d'entre elles par une variable duale (appliquer le principe de Lagrange).

$$L(w,b,\alpha) = \frac{1}{2}w^2 + \sum_{i=1}^{n}\alpha_i[y_i(< w, x_i > +b) - 1] \qquad (II.13)$$

Notons que L doit être minimisé par rapport aux variables primales w_i et *b* et maximisé par rapport aux variables duales $\boldsymbol{\alpha_i}$.

Le point selle (minimal par rapport à une variable, maximal par rapport à l'autre) doit donc satisfaire les conditions nécessaires de stationnarité (annule sa dérivé) qui correspondent aux conditions Karush Kuhn et Tucker (KKT) et de Fermat, nous trouvons:

$$\frac{\partial L(w,b,\alpha)}{\partial w} = 0 \qquad (II.14)$$

$$\frac{\partial L(w,b,\alpha)}{\partial b} = 0 \qquad (II.15)$$

Ce qui nous permet d'obtenir :

$$w = \sum_{i=0}^{n} \alpha_i y_i x_i \tag{II.16}$$

$$\sum_{i=0}^{n} \alpha_i y_i = 0 \tag{II.17}$$

Remarquons qu'avec cette formulation, on peut calculer *w* en fixant seulement *n* paramètres. L'idée va donc être de formuler un problème dual dans lequel *w* est remplacé par sa nouvelle formulation. De cette façon, le nombre de paramètres à fixer est relatif au nombre d'exemples de l'échantillon d'apprentissage et non plus à la dimension de l'espace d'entrée. Pour se faire, nous substituons (II.16) et (II.17) dans le Lagrangien (II.13), nous obtenons le problème dual équivalent suivant :

$$PQ2 \begin{cases} \underbrace{max}_{\alpha} \sum_{i=1}^{n} \alpha i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \\ SC \qquad \sum_{i=0}^{n} \alpha_i y_i = 0 \\ \qquad\quad \alpha_i \geq 0 \end{cases} \tag{II.18}$$

Ce dernier problème peut être résolu en utilisant des méthodes standards de programmation quadratiques. Une fois la solution optimale α^* du problème (II.18) obtenue, le vecteur de poids de l'hyperplan à marge maximale recherché s'écrit :

$$\mathrm{w}^* = \sum_{\mathrm{i=1}}^{\mathrm{N_{vs}}} \alpha_\mathrm{i}^* \mathrm{y_i}^* \mathrm{x_i}^* \qquad \text{avec} \quad \mathrm{N_{vs}} = \textit{nombre de vecteurs support} \tag{II.19}$$

Comme le paramètre *b* ne figure pas dans le problème dual, sa valeur optimale b^* peut être dérivée à partir des contraintes primales, soit donc :

$$b^* = -\frac{max_{\gamma i=-1}(<w^*, x_i>) + min_{\gamma i=1}(<w^*, x_i>)}{2} \tag{II.20}$$

Nous avons à présent tous les éléments nécessaires pour exprimer la fonction de décision de notre classificateur linéaire :

$$h(x) = sign(\sum_{i=1}^{nvs} \alpha i^* y_i^* \langle x, x_i^* \rangle + b^*) \tag{II.21}$$

Notons qu'un grand nombre de termes de cette somme est nul. En effet, seuls les α_i^* correspondants aux exemples se trouvant sur les hyperplans canoniques (sur la contrainte) sont non nuls. Ces exemples sont appelés Supports Vectors (**SV**). On peut les voir comme les

représentants de leurs catégories car si l'échantillon d'apprentissage n'était constitué que des **SV**, l'hyperplan optimal que l'on trouverait serait identique.

b. Marge souple ($(\exists\, \varepsilon_i \neq 0)$ *et* $C \geq 0$ *dans* $(PQ1)$)

Nous considérons ici le cas où certains exemples sont mal classés par l'hyperplan optimal. Cela peut résulter du bruit dans les données. Pour résoudre ce problème, Cortes et Vapnik en 1995 ont introduit la notion de « marge souple » (*soft margin*) qui correspond toujours à la recherche d'un hyperplan de marge optimale, mais avec une règle d'exception qui autorise que quelques exemples soient à une distance plus faible de l'hyperplan que la marge correspondante.

Soit $\varepsilon i = 1 - y_i . h(x_i)$ un indice mesurant l'importance de pénétration de l'exemple *xi* dans la zone définie par l'hyperplan H de marge géométrique *d*, $\varepsilon_i \neq 0$ pour $h(x_i) < 1$. Cette variable est appelée variable ressort (slack variable). S $\varepsilon_i > 1$i, l'exemple n'est pas du bon côté de l'hyperplan relativement à sa classe (exemple : $yi = 1, h(xi) = -1$ et $\varepsilon_i = 2 > 1$).

L'idée de la marge souple est de rechercher l'hyperplan de marge optimale pénalisée par l'importance des variables ressorts. Le terme de marge souple vient du fait que l'on peut considérer que les exemples pour les quels $\varepsilon_i > 0$, ont une marge géométrique réduite de $d(1 - \varepsilon_i)$. Le terme de pénalisation est de la forme $C \sum \varepsilon_i$ avec C une constante qui permet de définir l'importance de la pénalisation.

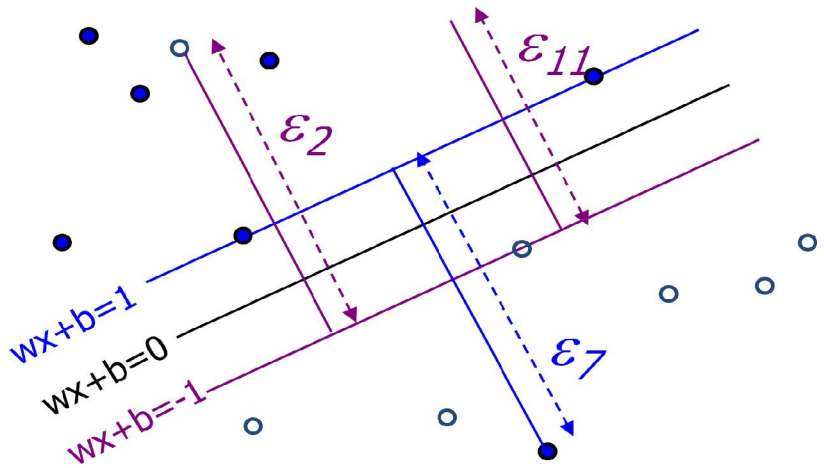


Figure II.10 : Marge souple et variable élastique $\boldsymbol{\varepsilon_i}$

Le paramètre ***C*** est défini par l'utilisateur. Il peut être interprété comme une tolérance au bruit du classifieur : pour de grandes valeurs de ***C***, seules de très faibles valeurs de $\boldsymbol{\varepsilon}$ sont autorisées, et par conséquent, le nombre de points mal classés sera très faible (données faiblement bruitées). Si ***C*** est petit, ε peut devenir très grand, et on autorise alors bien plus d'erreurs de classification (données fortement bruitées).

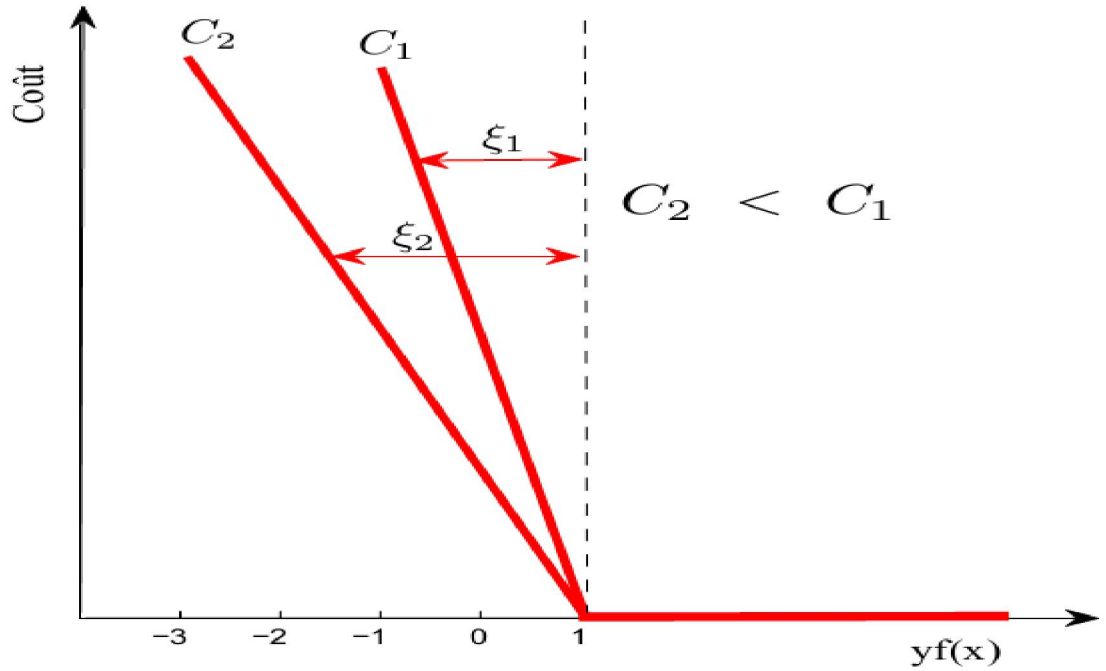


Figure II.11 : Représentation du compromis entre la tolérance C et la variable élastique ξi

En suivant la même démarche du Lagrangien que précédemment, nous aboutissons à la forme duale suivante :

$$L(w,b,\alpha,\varepsilon,\beta) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\varepsilon_i - \sum_{i=1}^{n}\alpha_i[y_i(\langle w,x_i\rangle + b) - 1 + \varepsilon_i] - \sum_{i=1}^{n}\beta_i.\varepsilon_i \quad \text{(II.22)}$$

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n}\alpha_i.y_i.x_i \quad \text{(II.23)}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n}\alpha_i.y_i = 0 \quad \text{(II.24)}$$

$$\frac{\partial L}{\partial \varepsilon} = C - \alpha_i - \beta_i = 0 \quad \text{(II.25)}$$

Ce qui nous permet d'obtenir :

$$w = \sum_{i=1}^{n} \alpha_i . y_i . x_i \quad \text{(II.26)}$$

$$\sum_{i=1}^{n} \alpha_i . y_i = 0 \quad \text{(II.27)}$$

$$\alpha_i = C - \beta_i \qquad 0 \leq \alpha_i \leq C \quad \text{(II.28)}$$

Ces conditions sont injectées dans (II.22) pour passer au problème dual :

$$\textbf{PQ3} \begin{cases} max \ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i . x_j \rangle & \text{(II.29)} \\ SC \ \sum_{i=1}^{n} \alpha_i y_i = 0 & \text{(II.30)} \\ 0 \leq \alpha \leq C \end{cases}$$

Pour résoudre ce problème ***(PQ3)*** il existe plusieurs algorithme tels que le SMO (Sequentiel Minimisation Optimisation)[10], SVM light[11], Simple SVM[12] afin de trouver les vecteurs de support.

II.3.2.2 Cas non linéairement séparable

Précédemment, nous avons décrit le principe des SVM dans le cas où les données sont linéairement séparables. Cependant, dans la plupart des problèmes réels, ce n'est pas toujours le cas et il est donc nécessaire de contourner ce problème (difficile de séparer n'importe quel jeu de données par un simple hyperplan). Si par exemple les données des deux classes se chevauchent sévèrement, aucun hyperplan séparateur ne sera satisfaisant.

L'idée est de projeter les points d'apprentissage dans un espace d'Hilbert H de dimension plus élevée dans lequel les données transformées devienne linéairement séparable et ce grâce à une fonction $\boldsymbol{\Phi}$ non-linéaire choisie a priori et d'appliquer la même méthode d'optimisation de la marge dans cet espace. L'espace ainsi obtenu est appelé espace des caractéristiques ou aussi espace transformé.

Le principe revient donc à résoudre les problèmes ***PQ2*** et ***PQ3*** dans l'espace H, en remplaçant $\langle \boldsymbol{xi}.\boldsymbol{xj} \rangle$ par $\langle \boldsymbol{\Phi}(\boldsymbol{xi}).\boldsymbol{\Phi}(\boldsymbol{xj}) \rangle$.

Le problème quadratique obtenu peut s'écrire comme suit :

$$PQ4 \begin{cases} max \ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \Phi(x_i). \Phi(x_j) \rangle & \text{(II.31)} \\ SC \ \sum_{i=1}^{n} \alpha_i y_i = 0 & \text{(II.32)} \\ 0 \leq \alpha_i \leq C \end{cases}$$

a. Astuce de noyau

Vu que les données apparaissent dans tous les calculs uniquement sous forme de produits scalaires $\langle \Phi(x_i). \Phi(x_j) \rangle$, il suffit de faire appel à une fonction noyau $K(x_i, x_j)$ qui permet ce calcul $K(x_i, x_j) = \langle \Phi(x_i). \Phi(x_j) \rangle$ et qui satisfait la condition de Mercer.

Condition de Mercer

On dit d'une fonction est un noyau si et seulement si la condition suivante est vérifiée :

$G = K(x_i, x_j))_{i,j=1}^{n}$ est défini positive autrement dit elle vérifie les trois propriétés fondamentales du produit scalaire :

- Positivité : $K(x_i, x_j) \geq 0$
- Symétrie : $K(x_i, x_j) = K(x_j, x_i)$
- Inégalité de Cauchy-Shwartz : $K(x_i, x_j) \leq \|x_i\|. \|x_j\|$

G est une matrice contenant les similarités entre tous les exemples de l'ensemble d'apprentissage appelée matrice de Gram, elle a une importance cruciale dans les algorithmes à noyaux car c'est elle qui définit la complexité numérique de l'apprentissage ; pour le problème de la classification SVM, elle permet de définir la partie quadratique de la forme quadratique à optimiser et elle contient aussi toutes les informations sur les données d'apprentissage et la fonction *K*.

Parmi toutes les fonctions noyaux utilisées pour répondre aux besoins des SVM, on cite les plus utilisées :

Le noyau linéaire : est un simple produit scalaire : $K(x_i, x_j) = \langle x_i, x_j \rangle$ (II.33)

Le noyau polynomial : permet de représenter des frontières de décision par des polynômes de degré *d* : $K(x_i, x_j) = (a * \langle x_i, x_j \rangle + b)^d$ (II.34)

La dimension de l'espace transformé induit par un noyau polynomial est de l'ordre $\frac{(p+d)!}{p!d!}$, où *p* est la dimension de l'espace de départ.

Le noyau gaussien ou RBF (Radial Basis Function) : qui a la forme suivante :

$$K(x_i, x_j) = exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad \text{(II.35)}$$

Le paramètre σ permet de régler la largeur de la gaussienne. En prenant un σ grand, la similarité d'un exemple par rapport à ceux qui l'entourent sera assez élevée, alors qu'on prenant un σ tendant vers 0, l'exemple ne sera similaire à aucun autre.

b. Schéma de fonctionnement général avec les fonctions noyaux

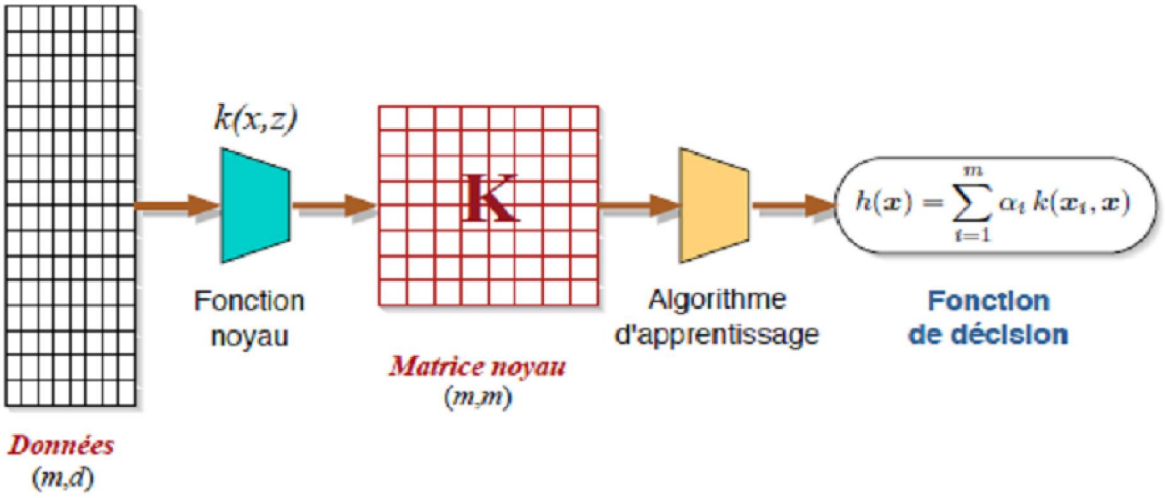


Figure II.12 : Chaîne de traitements génériques d'une méthode à noyau

II.4 Le choix des paramètres optimaux

La réalisation d'un programme d'apprentissage par SVM se ramène à résoudre un problème d'optimisation impliquant un système de résolution dans un espace de dimension conséquente. L'utilisation de ces programmes revient surtout à sélectionner une bonne famille de fonctions noyau et à régler les paramètres de ces fonctions. Ces choix sont le plus souvent faits par une technique de validation croisée, dans laquelle on estime la performance du système en la mesurant sur des exemples n'ayant pas été utilisés en cours d'apprentissage.

L'idée est de chercher les paramètres permettant d'obtenir la performance maximale. Si la mise en œuvre d'un algorithme de SVM est en général peu coûteuse en temps, il faut cependant compter que la recherche des meilleurs paramètres peut requérir des phases de test assez longues.

II.5 Extension des SVM

La plupart des problèmes ne se contentent pas de deux classes de données. Il existe plusieurs méthodes pour faire la classification multi-classes.

La première méthode est appelé Un-Contre-Tous. C'est une approche étendant la notion de marge aux cas multi-classes. Cette formulation intéressante permet de poser un problème d'optimisation unique. Le problème fait intervenir N fonctions de décision.

La deuxième méthode est une méthode dite Un-contre-Un. Au lieu d'apprendre N fonctions de décisions, ici chaque classe est discriminée d'une autre.

II.5.1 Approche Un-contre-Tous (1vsR)

L'idée de cette stratégie est de construire autant de classificateurs que de classes. Ainsi, durant l'apprentissage, tous les exemples appartenant à la classe considérée sont étiquetés positivement (+1) et tous les exemples n'appartenant pas à la classe sont étiquetés négativement (-1).

Un hyperplan H_k est défini pour chaque classe k par la fonction de décision suivante :

$$H_k(x) = sign(\langle w_k, x\rangle + b_k) \tag{II.36}$$

$$= \begin{cases} +1, si\ H_k(x) > 0 \\ -1, sinon \end{cases}$$

$$k^* = Arg_{1\le k\le K}\, Max(H_k(x)) \tag{II.37}$$

Si une seule valeur $H_k(x)$ est égale à 1 et toutes les autres sont égales à -1, on conclut que x appartient à la classe k. Or, il est possible que plusieurs sorties soient positives pour un exemple de test donné. Ceci est particulièrement le cas des données ambiguës situées près des frontières de séparation des classes. On utilise dans ce cas un vote majoritaire pour attribuer l'exemple x à la classe k.

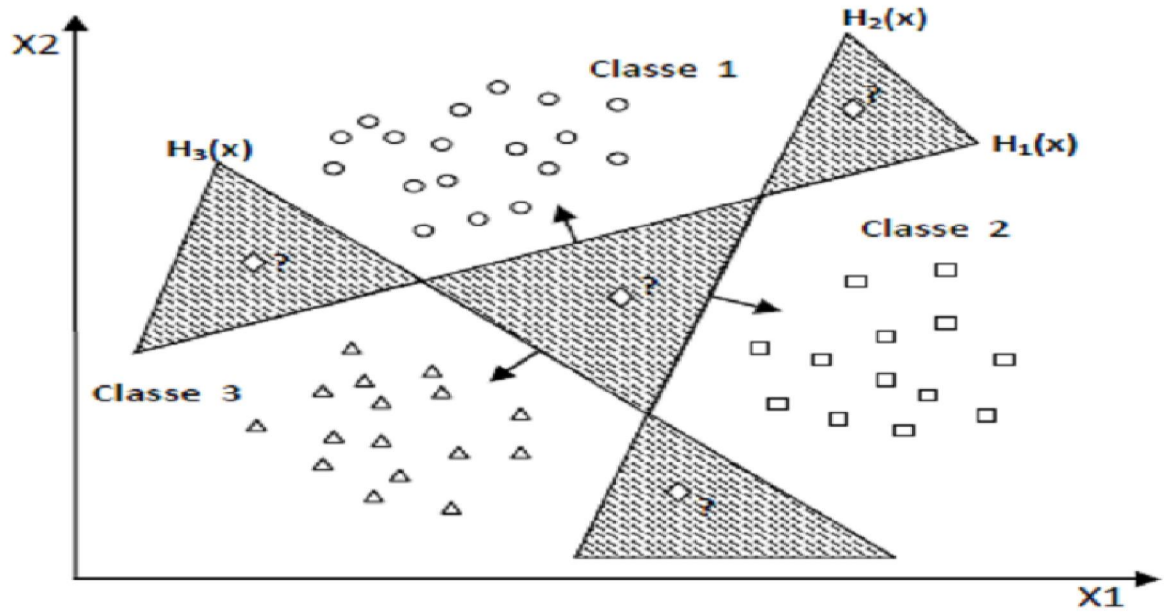


Figure II.13 : Approche un contre tous

Cette méthode est critique à cause de son asymétrie, puisque chaque hyperplan est entrainé sur un nombre d'exemples négatifs beaucoup plus important que le nombre d'exemples positifs. La méthode un contre un est une méthode symétrique qui corrige ce problème.

II.5.2 Approche Un-contre-Un (1vs1)

L'approche Un-Contre-Un est un cas spécial des méthodes de décomposition proposées par Dietterich et al. [13] pour résoudre des problèmes à plusieurs classes. Cette approche requiert la construction de K (K − 1)/2 SVM chacun séparant un couple de classes *(i, j)* parmi ceux existants.

Pendant la classification, un vecteur d'entrée x est présenté à l'ensemble des classificateurs construits. La sortie de chaque SVM fournit un vote partiel concernant uniquement le couple de classes (*wi,wj*). En considérant que chaque SVM calcule un estimé $\hat{p}_{ij}$ de la probabilité :

$$p_{ij} = P(x \in w_i | x, x \in w_i \cup w_j) \tag{II.38}$$

alors la règle de classification la plus simple peut s'écrire :

$$argmax_{1 \le i \le k} \sum_{j \neq i} [\hat{p}_{ij} > 0.5] \tag{II.39}$$

L'opérateur [ ] est défini :

$$[\eta] = \begin{cases} 1 \; si \; \eta \; est \; vrai \\ 0 \; sinon \end{cases} \tag{II.40}$$

Cette combinaison considère que les sorties des SVM sont des valeurs binaires de 1 ou 0. Une autre approche de reconstruction pourrait tirer avantage de l'information de confiance associée à chacune des sorties$\hat{p}_{ij}$. Dans l'hypothèse que ces valeurs représentent des probabilités, il est possible d'estimer une approximation $\hat{p}_i$ de la probabilité à posteriori :

$$p_{ij} = P(x \; \epsilon \; w_i | x) \tag{II.41}$$

En considérant la matrice carrée $\hat{P}$ avec les entrées $\hat{p}_{ij}$ tels que $(i,j)_{i,j=1,\dots k.}$ avec $\hat{p}_{ji} = 1 - \hat{p}_{ij}$. Les valeurs de $\hat{p}_i$ peuvent être calculées par :

$$\hat{p}_i = \frac{2}{K(K-1)} \sum_{i \neq j} \hat{p}_{ij} \tag{II.42}$$

Et la règle de décision s'écrit :

$$\arg\max_{1 \leq i \leq K} \hat{p}_i \tag{II.43}$$

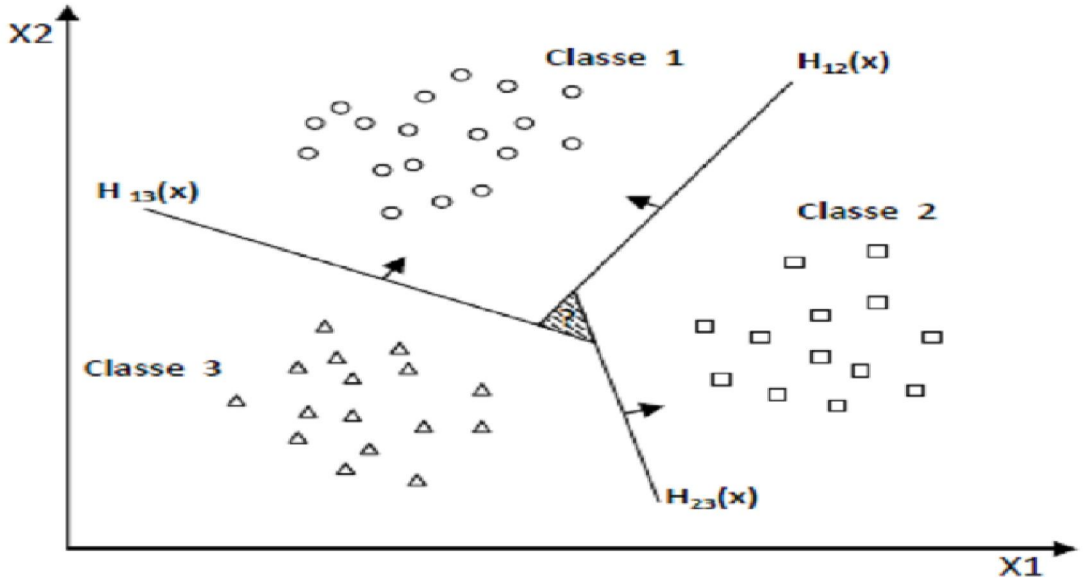


Figure II.14 : Approche un contre un[5].

II.6 Conclusion

Dans ce chapitre, nous avons présenté de manière simple et complète le concept de système d'apprentissage introduit par Vladimir Vapnik, les « Support Vector Machine ». Nous avons donné une vision générale et une vision purement mathématiques des SVM. Cette méthode de classification est basée sur la recherche d'un hyperplan qui permet de séparer au mieux des ensembles de données. Nous avons exposé les cas linéairement séparable et les cas non linéairement séparables qui nécessitent l'utilisation de fonction noyau (kernel) pour changer d'espace.

L'utilisation de ce classifieur pour reconnaitre des extrasystoles ventriculaires nécessite que chaque battement cardiaque « i » soit d'une part quantifié par un vecteur "x_i" de dimension p et d'autre part qu'à chaque battement « i » soit associée une étiquette "y_i" car il s'agit d'un classifieur supervisé.

Dans ce qui suit, les battements cardiaques seront localisés de façon automatique puis caractérisés et étiquetés. Pour résoudre le problème quadratique, l'algorithme utilisé est basé sur la méthode des contraintes actives [7].

Chapitre III

Caractérisation de battements cardiaques

III.1 Introduction

Dans l'application envisagée, nous considérons deux classes de battements cardiaques : la classe normale et la classe extrasystolique. Il faudrait que nous arrivions à les localiser puis à les caractériser par les paramètres les plus discriminants possibles. L'outil de traitement du signal requis est l'analyse multirésolution car elle nous permet de travailler sur différentes bandes de fréquence.

Avant ceci, nous avons jugé nécessaire de donner un aperçu sur le signal ElectroCardioGraphique (ECG) de façon générale, son aspect et ses ondes d'activation. Après ceci, les battements normaux et pathologiques seront mis en évidence.

III.2 Notions d'Electrocardiographie

III.2.1 Activité électrique du cœur

Lorsqu'on parle du fonctionnement électrique du cœur, il faut revenir au niveau cellulaire et se rappeler qu'il existe une polarisation naturelle de la cellule. Lorsque la cellule est stimulée électriquement, les propriétés de la membrane sont modifiées et sa perméabilité aux ions augmente.

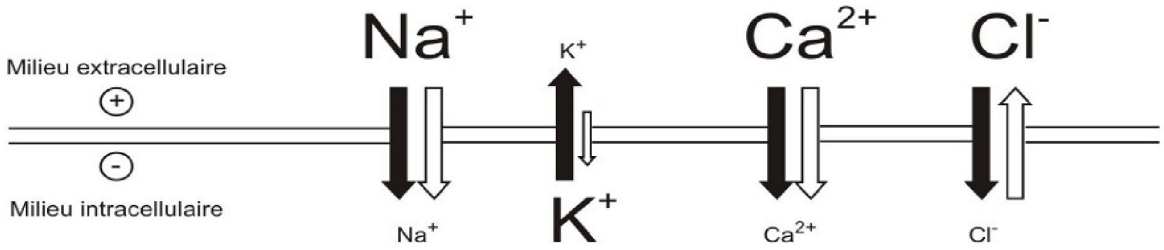


Figure III.1 : Les échanges ioniques membranaires.

Les échanges ioniques à travers la membrane des cellules donnent naissance au potentiel d'action.

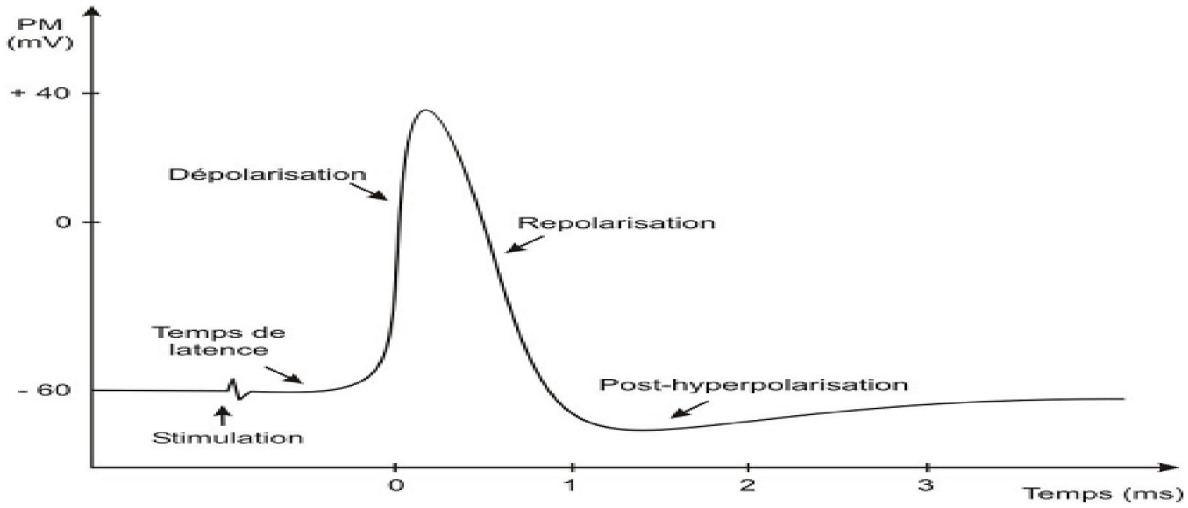


Figure III.2 : Le potentiel d'action [14].

L'activité électrique cardiaque est un stimulus généré automatiquement, il prend naissance dans le nœud sinusal (NS) puis se propage selon un cheminement : nœud sinusal, myocarde auriculaire, nœud auriculo-ventriculaire d'Aschoff-Tawara, faisceau de His et ses branches gauche et droite, réseau sous-endocarditique de Purkinje, myocarde ventriculaire.

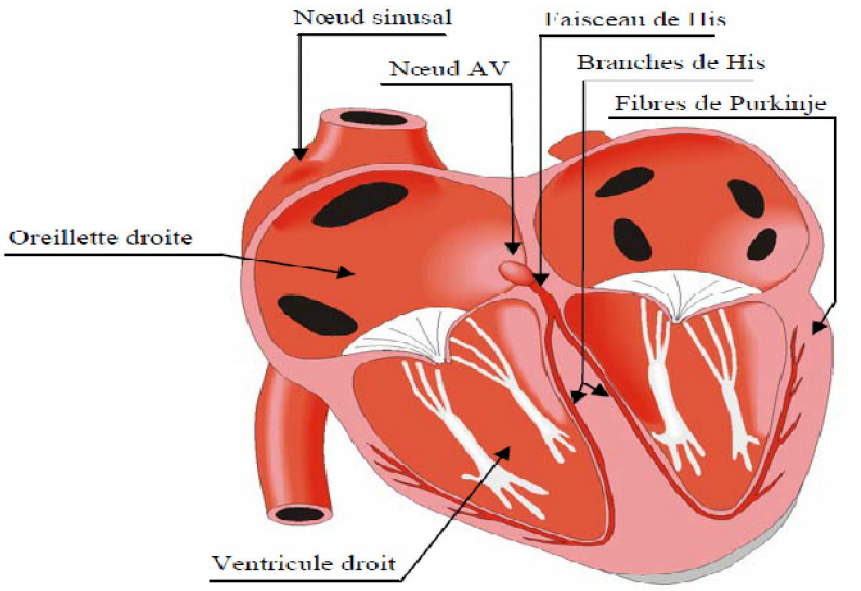


Figure III.3 : Cheminement de l'activité électrique du cœur.

III.2.2 L'Electrocardiogramme

L'ECG est l'enregistrement de l'activité électrique du cœur. Cette activité correspondant à la dépolarisation et à la repolarisation du myocarde. L'ECG se présente

comme une suite de déflexions (ondes) correspondant chacune à une phase de fonctionnement du cœur.[7]

En isolant un battement cardiaque normal, on observe l'existence de trois ondes élémentaires : onde P, complexe QRS et l'onde T. L'onde P correspond à la dépolarisation des oreillettes. Elle est suivie de la dépolarisation des ventricules, qu'on appelle le complexe QRS. La repolarisation des ventricules est représentée par l'onde T (voir figure III.4).

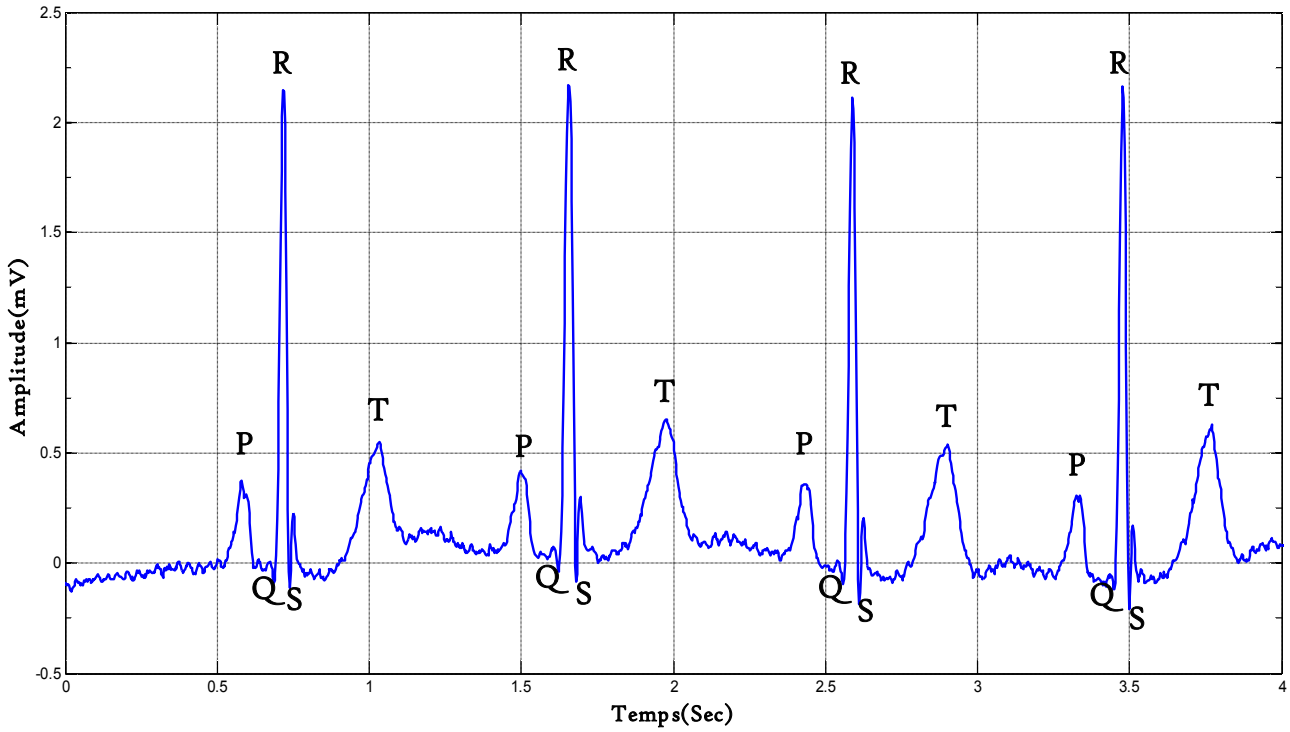


Figure III.4 : Allure d'un électrocardiogramme normal.

III.3 Les extrasystoles ventriculaires

Les extrasystoles ventriculaires (ESV) sont des battements anormaux. Ils représentent l'arythmie, la plus fréquente. Ils s'observent sur quasiment tous les enregistrements, principalement en période de récupération après un effort. Contrairement aux battements normaux qui ont pour origine la dépolarisation des cellules sinusales, l'ESV naît de la dépolarisation spontanée d'un petit groupe de cellules ventriculaires, appelé alors foyer ectopique ventriculaire. L'impulsion électrique créée n'emprunte pas la voie normale de conduction (faisceau de His). Elle se propage alors lentement dans les ventricules. La contraction ventriculaire ainsi étalée dans le temps perd de son efficacité.

III.3.1 Caractéristiques morphologiques des ESV

Sur un tracé ECG, un battement ESV est caractérisé par les propriétés suivantes:

- ✓ l'onde R n'est pas précédée d'une onde P, puisqu'il n'y a pas eu d'activité auriculaire préalable,
- ✓ La durée du complexe QRS est supérieure à la durée d'un complexe QRS normal car l'impulsion électrique n'emprunte pas la voie normale de conduction et se propage donc lentement dans les ventricules.
- ✓ La forme du complexe QRS est atypique.
- ✓ Ces battements sont prématurés.

Un exemple d'ECG contenant des battements ESV est présenté en figure III.5.

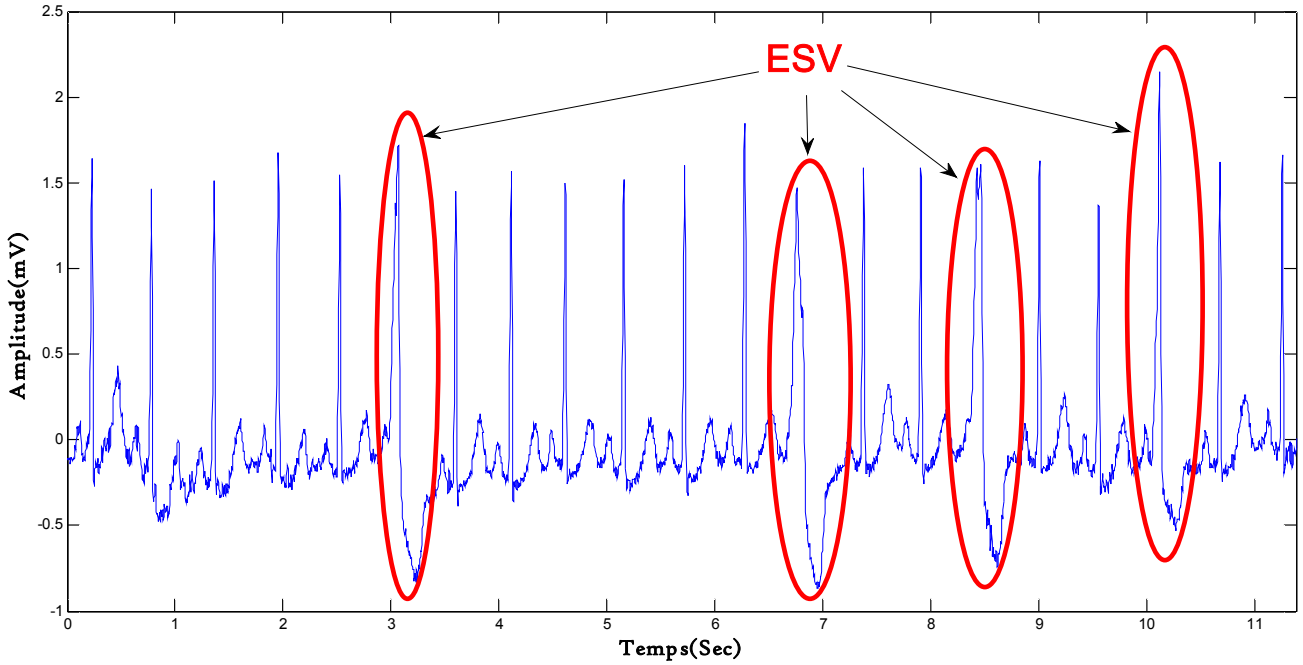


Figure III.5 : ECG Avec extrasystoles ventriculaires.

III.3.2 Contenu fréquentiel des battements

Le signal ECG est un signal riche par la variété des ondes qui le constitue. Ainsi, il présente une densité spectrale de puissance qui varie en fonction de la morphologie du signal et d'un sujet à un autre. La densité spectrale de puissance des signaux ECG a été étudiée dans tous les cas possible et il a été prouvé que la densité spectrale de puissance des complexes QRS se situe dans la bande de fréquence de 0 à 30 Hz.

L'origine physiologique d'un battement extrasystolique fait qu'il possède une morphologie plus ample qu'un battement normal. Ceci nous amène à examiner le contenu fréquentiel des différents types de complexes QRS.

Pour analyser le contenu fréquentiel des complexes QRS, nous avons sélectionné les deux types de battement considérés : les battements sinusaux et les ESV. Les complexes QRS sont extraits grâce à une fenêtre de longueur 180 ms autour des positions annotées des ondes R. La densité spectrale de puissance (DSP) de ces segments est alors calculée en respectant le théorème de Vienner-Kinchine. Les résultats moyens obtenus sont illustrés par la figure III.6.

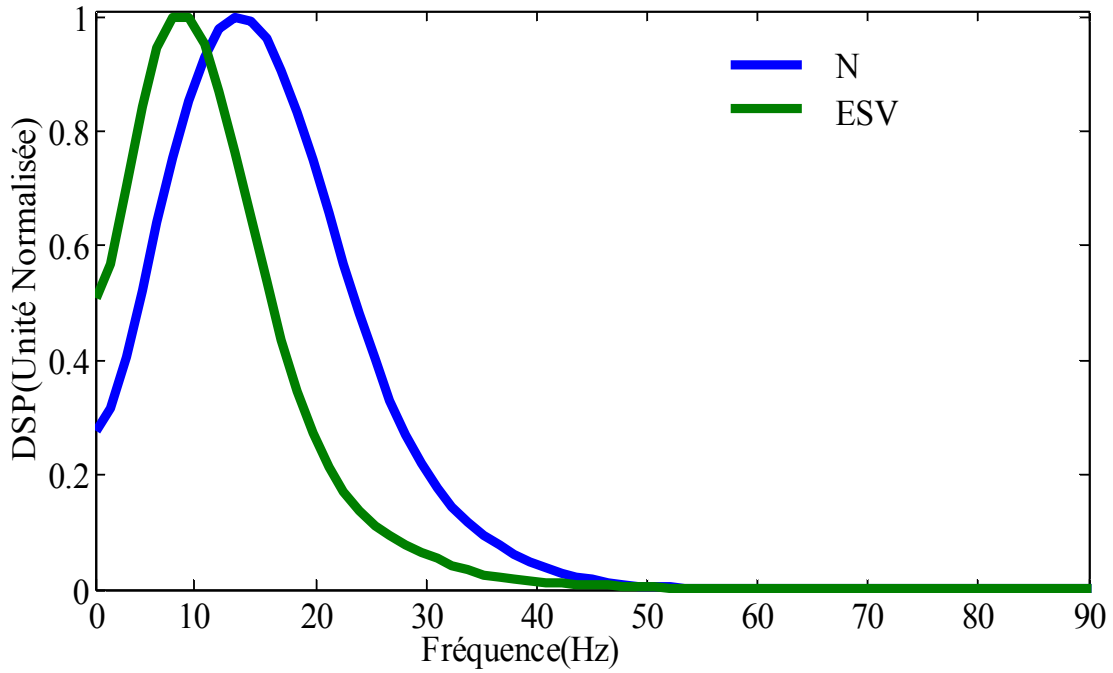


Figure III.6 DSP des battements N et ESV

III.4 Outil de traitement requis

III.4.1 Choix de l'outil de traitement

La figure III. 6 montre qu'un battement ESV est légèrement décalé vers les basses fréquences par rapport à un battement normal. La projection du signal sur différents niveaux de résolution va surement nous permettre de discriminer les deux types de battements. Une analyse multirésolution serait alors la plus adéquate.

D'autre part, notre base d'étude est constituée de signaux fortement bruités et non stationnaires, surtout en phase d'arythmie. En plus des artefacts qui sont d'origines et de natures diverses, la base d'étude présente une grande variabilité de la forme du signal, d'un

sujet à un autre, et même chez le même sujet, d'un cycle à un autre. l'analyse multirésolution nous permet l'analyse du signal sur différentes bandes de fréquences [7].

III.4.2 Analyse multirésolution

Le principe de base de l'Analyse MultiRésolution (AMR) consiste à séparer de façon itérative, le signal en deux composantes, l'une représentant l'allure du signal (approximation) et l'autre, ses détails. Cette opération est réalisée grâce à la projection du signal sur deux sous espaces vectoriels orthogonaux et complémentaires, l'un appelé espace des approximations et l'autre espace des détails.[7]

III.4.2.1 La fonction ondelette

Nous commençons d'abord par définir l'ondelette ψ, une fonction à support compact, oscillante, de moyenne nulle et de carré sommable. On définit une famille $\psi_{a,b}$ d'ondelettes à partir d'une ondelette mère ψ par sa dilatation ou compression et sa translation :

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \quad \text{(III.1)}$$

Où a est le facteur d'échelle (dilatation, compression) et b, le facteur de translation, paramètre de localisation temporelle. La pondération en $\frac{1}{\sqrt{a}}$ permet d'avoir des fonctions analysantes de même norme : toutes les ondelettes de la même famille ont une même énergie.

$$\forall a > 0, \forall b \in \mathbb{R}, \int_{-\infty}^{+\infty} \left|\psi_{a,b}(t)\right|^2 dt = \|\psi\|^2 = 1 \quad \text{(III.2)}$$

Dans l'expression $\psi\left(\frac{t-b}{a}\right)$, le pas de translation à l'échelle a est b/a. On pose :

$$a = a_o^j \; et \; b = nb_o a_o^j \quad \text{(III.3)}$$

avec a_o,$b_o \in \mathbb{Z}$. Si on choisit $a_o = 2 \; et \; b_o = 1$, on parle alors de fonction ondelette dyadique. Pour un niveau de résolution j, on a :

$$\psi_{j,n}(t) = 2^{-\frac{j}{2}}\psi(2^{-j}t - n) \quad \text{(III.4)}$$

Les fonctions $\psi_{j,n}$ forment alors une base orthonormée par translation et dilatation ou compression :

Soit $\psi_{0,0}(t)$, la fonction ondelette au niveau 0. On a :

$$\psi_{0,0}(t) = \psi(t) \begin{cases} \psi_{j,n}(t) = 2^{-\frac{j}{2}}\psi(2^{-j}t - n) \quad (dilatation\ de\ rapport\ 2^j) \\ \psi_{j,n}(t) = 2^{\frac{j}{2}}\psi(2^{j}t - n) \quad (compression\ de\ rapport\ 2^j) \end{cases} \tag{III.5}$$

III.4.2.2 La fonction d'échelle

L'apparition de l'analyse multirésolution coïncide avec l'introduction d'une seconde fonction appelée : fonction d'échelle φ : le père des ondelettes telle que la famille :

$$\varphi_{j,n} \cup \psi_{j,n}, j \geq 0\ n \in \mathbb{Z} \tag{III.6}$$

Forme une base orthonormée. Les fonctions $\varphi_{j,n} \in L^2(\mathbb{R})$ sont construites suivant la relation :

$$\varphi_{j,n}(t) = 2^{-\frac{j}{2}}\varphi(2^{-j}t - n) \tag{III.7}$$

La fonction ondelette et la fonction d'échelle sont étroitement liées. A chaque niveau de résolution, la fonction ondelette est une combinaison linéaire de sa fonction d'échelle.

III.4.2.3 Décomposition du signal

La décomposition d'un signal sur un niveau de résolution consiste en sa projection sur deux sous espaces vectoriels :

a. Espace d'approximations

Nous nous plaçons dans l'espace$L^2(\mathbb{R})$ de fonctions continues à variable réelle et à énergie finie. Une analyse à la résolution j d'une fonction *f(t)* est obtenue par action d'un projecteur linéaire A_j sur *f(t)*. L'approximation de *f(t)* sur ce niveau est :

$$a_j(t) \in V_j$$

V_j étant un sous espace de L^2. On construit une analyse multirésolution à l'aide de sous espaces V_j emboités tels que le passage de l'un à l'autre soit le résultat d'un changement d'échelle. Par exemple dans le cas dyadique :

$$a_j(t) \in V_j \Leftrightarrow a_j\left(\frac{t}{2}\right) \in V_{j+1}$$

Ce qui correspond à une dilatation d'un facteur 2. L'espace V_{j+1} contient des signaux de plus basse fréquence que V_j, autrement dit : $V_{j+1} \subset V_j$. Pour un sous ensemble de L^2, on a :

$\ldots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \ldots \subset V_{j+1} \subset V_j \subset \cdots$ avec $j \in \mathbb{Z}$.

Si la fonction d'échelle $\varphi_{j,n}$ engendre une base orthonormée V_j, la projection par A_j d'une fonction *f(t)* sur cette base fournira les coefficients de cette décomposition. Ces coefficients qui décrivent l'approximation de *f(t)* à l'échelle *j* sont appelés coefficients d'échelle et sont donnés par :

$$a_n^j = < f(t), \varphi_{j,n}(t) > \quad \text{(III.8)}$$

Et

$$a_j(t) = \sum_n < f(t), \varphi_{j,n}(t) > \varphi_{j,n}(t) = \sum_n a_n^j \, \varphi_{j,n}(t) \quad \text{(III.9)}$$

La base étant orthonormée,

$$\left\| a_j(t) \right\|^2 = \sum_{n=-\infty}^{+\infty} \left| a_n^j \right|^2 \quad \text{(III.10)}$$

b. Espace de détails

On vient de voir que $V_j \subset V_{j-1}$, on peut alors définir pour chaque V_j, son complément orthogonal W_j dans V_{j-1} qui nous permet de récupérer les détails perdus en passant de V_{j-1} à V_j tel que :

$$V_{j-1} = V_j \oplus W_j \quad \text{(III.11)}$$

Comme W_{j-1} est orthogonal à V_{j-1}, W_{j-1} est orthogonal à W_j. Cette propriété s'écrit :

$$W_j \perp W_k \, , \forall j \neq k$$

La base W_j est engendrée par la fonction ondelette $\psi_{j,n}$. L'approximation à l'échelle immédiatement plus fine pourra être reconstruite en utilisant les détails du signal fournis par sa projection sur la base W_j suivant la relation :

$$a_{j-1}(t) = a_j(t) \sum_n < f(t), \psi_{j,n}(t) > \psi_{j,n}(t) = \sum_n a_n^j \, \psi_{j,n}(t) = a_j(t) + d_j(t) \quad \text{(III.12)}$$

$d_j(t)$ est la projection de *f(t)* sur W_j. Le signal détail est décrit par les coefficients de détails notés : $d_j(t) = < f(t), \psi_{j,n}(t) >$ tel que :

$$d_j(t) = \sum_n d_n^j \psi_{j,n}(t) \qquad \text{(III.13)}$$

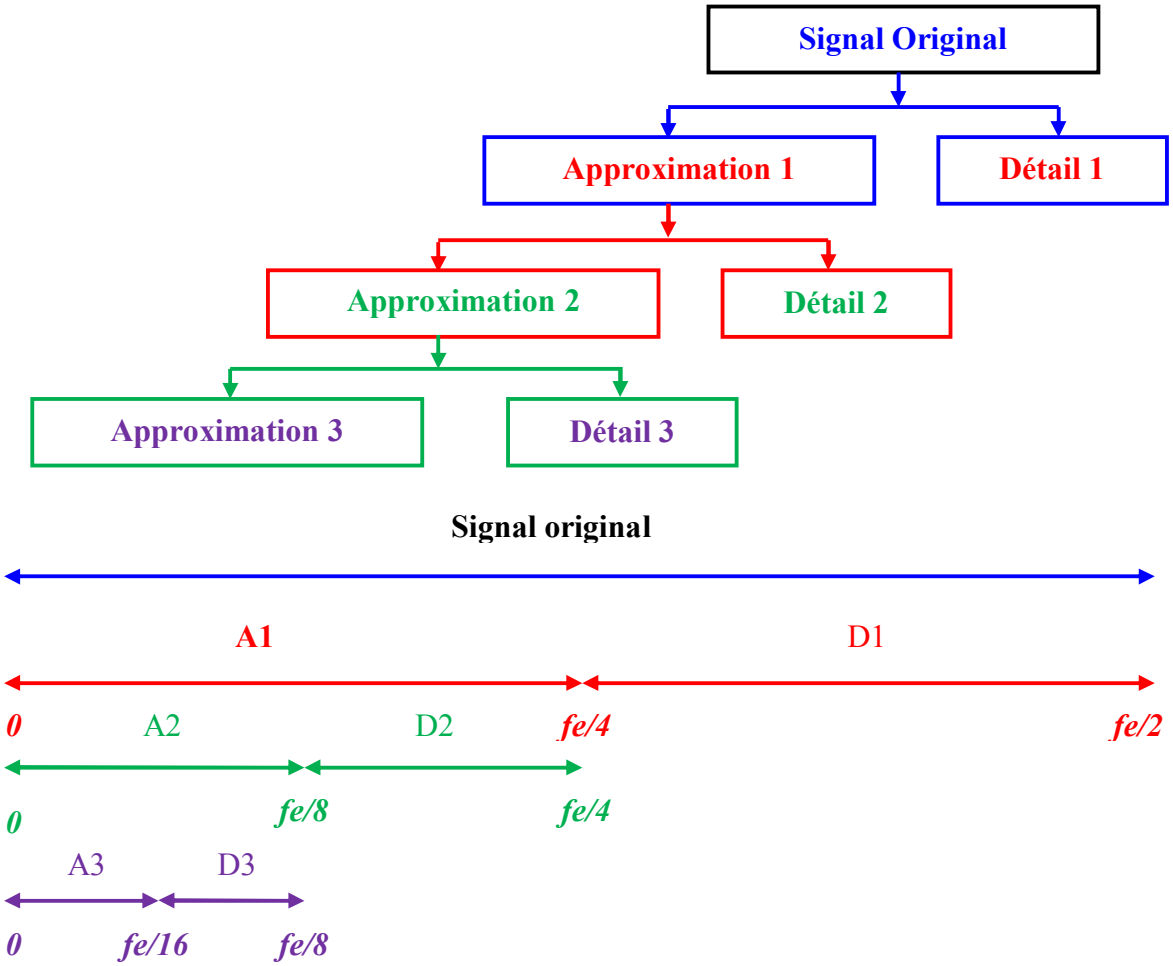


Figure III.7 : Décomposition d'un signal au niveau de résolution N.

III.5 Filtrage du signal

Chaque signal capté comporte un signal utile et du bruit. Le filtrage est une étape de grande importance pour la suppression du bruit qui s'ajoute au signal utile. Divers bruits sont présents dans l'ECG de routine. Parmi les plus importants, on peut citer :

- La dérive de la ligne de base.
- Les artefacts dus aux mouvements.
- Les tremblements musculaires

Le signal ECG étant un signal multi-composantes, non stationnaire affecté par un bruit dont certaines composantes comme le bruit musculaire sont corrélées. La représentation temps-échelle s'avère un outil plus adapté à son traitement. Nous utilisons le seuillage des coefficients de détails après la décomposition du signal par analyse multirésolution.
Les coefficients en dessous du seuil, correspondent à du bruit et peuvent être supprimés. La décomposition jusqu'au niveau 3 a été choisie dans le but de ne pas dégrader les complexes QRS après le seuillage, c'est pour supprimer les fluctuations du signal. Cette méthode nous aide à nous débarrasser des hautes fréquences mais pas des basses fréquences qui font que la ligne de base du tracé ECG soit ondulée. Nous pouvons les éliminer par un filtrage passe haut vu que cette ondulation est de très basse fréquence et que notre complexe QRS appartient à une bande de fréquence supérieur à 5 Hz. En annulant l'élément de plus basse fréquence, les ondulations vont disparaitre et le signal ECG sera sur la ligne de base.

Parmi les avantages qu'offre cette méthode est que sur la même décomposition, on effectue le seuillage des coefficients d'ondelettes puis on annule la dernière approximation. On réalise à la fois le filtrage HF et la suppression des ondulations de la ligne de base. Un exemple de résultats de l'opération de filtrage est présenté sur la figure suivante.

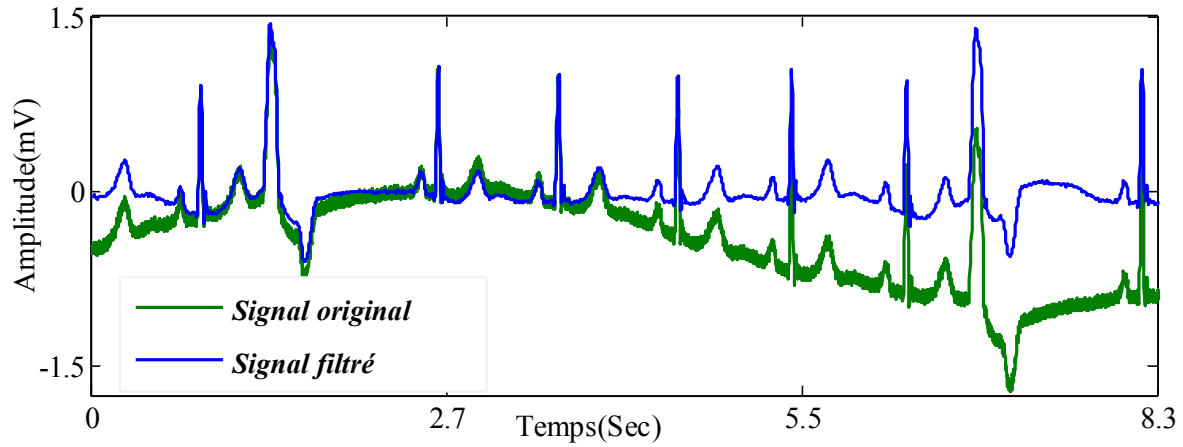


Figure III.8 : Exemple de filtrage.

III.6 Segmentation des battements cardiaques

Segmenter le signal ECG consiste donc à le partitionner en trames successives et homogènes définissant chacune un cycle cardiaque. Dans notre cas, l'objectif est la segmentation des intervalles QRS afin de les quantifier.

III.6.1 Détection de l'onde R

Cette opération est incontournable dans toute analyse automatique de l'ECG car l'onde R est la plus pertinente dans le signal. Elle nous permet alors de nous positionner sur le cycle cardiaque.

Cette détection semblerait pouvoir être effectuée par un simple seuillage du signal, car les ondes R sont en général de plus grande amplitude que les autres. Mais ce n'est pas le cas ; parfois, l'onde T est d'amplitude comparable à celle de R, ce qui pourrait être une sérieuse cause d'erreur. Une bonne détection des complexes QRS nécessite donc un traitement du signal plus élaboré.

L'algorithme de détection que nous utilisons dans cette étude est celui proposé dans [7]

Sélection des bandes de fréquence

Nous utilisons l'analyse multirésolution pour décomposer le signal en différentes bandes de fréquences, afin que les hautes fréquences et les basses fréquences puissent être analysées séparément. Les niveaux de décomposition sont fixés en fonction de la fréquence d'échantillonnage et de la bande de fréquence recherchée. Dans notre cas, nous avons choisi le niveau 5 en nous basant sur les densités spectrales de puissance des complexes QRS et sur la fréquence d'échantillonnage qui est de 360 Hz. En tenant compte du théorème de Shannon, la bande de fréquence du signal est [0 Hz, 180 Hz] où 180 Hz représente la fréquence de Nyquist. La correspondance entre les coefficients de détail obtenus et les bandes de fréquences est représentée sur la table.

.

Détail	Bande de fréquence
d_n^1	[90 Hz, 180 Hz]
d_n^2	[45 Hz, 90 Hz]
d_n^3	[22.5 Hz, 45 Hz]
d_n^4	[11.25 Hz, 22.5 Hz]
d_n^5	[5.62 Hz, 11.25 Hz]

Tableau 1 : Décomposition en 5 niveaux de résolution.

Par ailleurs, le résultat présenté en figure III.6 montre que l'énergie des deux types de complexes QRS est essentiellement concentrée entre 5Hz et 22Hz c'est-à-dire sur les détails 4 et 5. Ces deux détails sont alors retenus dans l'algorithme de détection.

La densité spectrale de puissance correspondant aux niveaux 4 et 5 sont représentée en figure *III.9*. Les niveaux 1 et 2 ne sont pas considéres car ils sont en dehors de la bande d'intérêt. La figure III.10 montre que les battements normaux sont plus significatifs sur le niveau de résolution 4 tandis que les battements PVC sont plus significatifs sur le niveau 5. Les coefficients de détail 4 et 5 sont donc sélectionnés pour mettre au point notre algorithme de détection. Leur produit h est utilisé pour localiser les complexes QRS. Ainsi, les ondulations de la ligne de base et l'effet de l'onde T sont supprimé

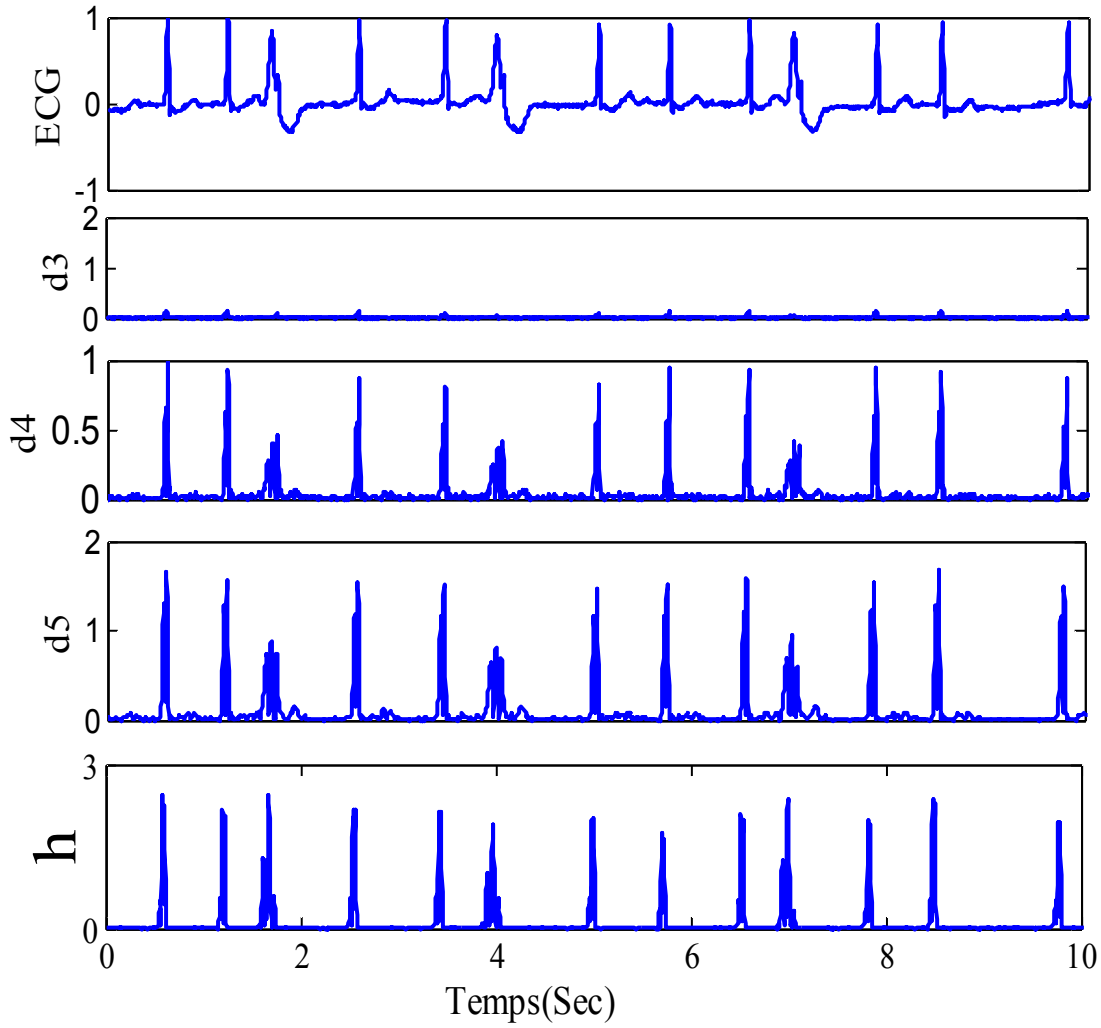


Figure III.9 Exemple de décomposition du signal (les détails 1 et 2 ainsi que l'approximation A5 ne sont pas présentés car ne contiennent aucune énergie provenant des QRS)

Algorithme

1. Décomposer le signal ECG *f(t)jusqu'à 5 niveau de résolution.*
2. Calculer $h = \left|\prod_{j=4}^{5}(d_n^j)\right|$.
3. Localisation des QRS :

$$\begin{cases} si\ h(n) \geq 0{,}3.max(h)\ alors\ n \implies QRS\ candidat \\ sinon\ n \nRightarrow QRS \end{cases}$$

4. Soit *n* et *n' deux positions consécutives sélectionnées, alors :*

$$\begin{cases} si\ |n - n'| < 36\ alors\ n\ et\ n' \implies même\ QRS \\ sinon\ n\ et\ n' \nRightarrow même\ QRS \end{cases}$$

   (36/f_e=100 ms est la durée standard d'un complexe QRS)
5. Une détection multiple dans un intervalle de 200 ms doit être supprimée. Cette contrainte a un sens physiologique : la période réfractaire est nettement supérieure à 200 ms.
6. Recherche des battements omis : Si aucun battement n'est détecté dans une période égale à 1.5 fois l'intervalle RR courant, le seuil de détection est divisé par 2 pour rechercher un éventuel QRS omis. Cette période de recherche a un sens physiologique : un inter-battement ne change pas aussi vite.[7]

La figure III.10 montre un exemple de détection des pics R. Davantage de résultats seront présentés au chapitre suivant.

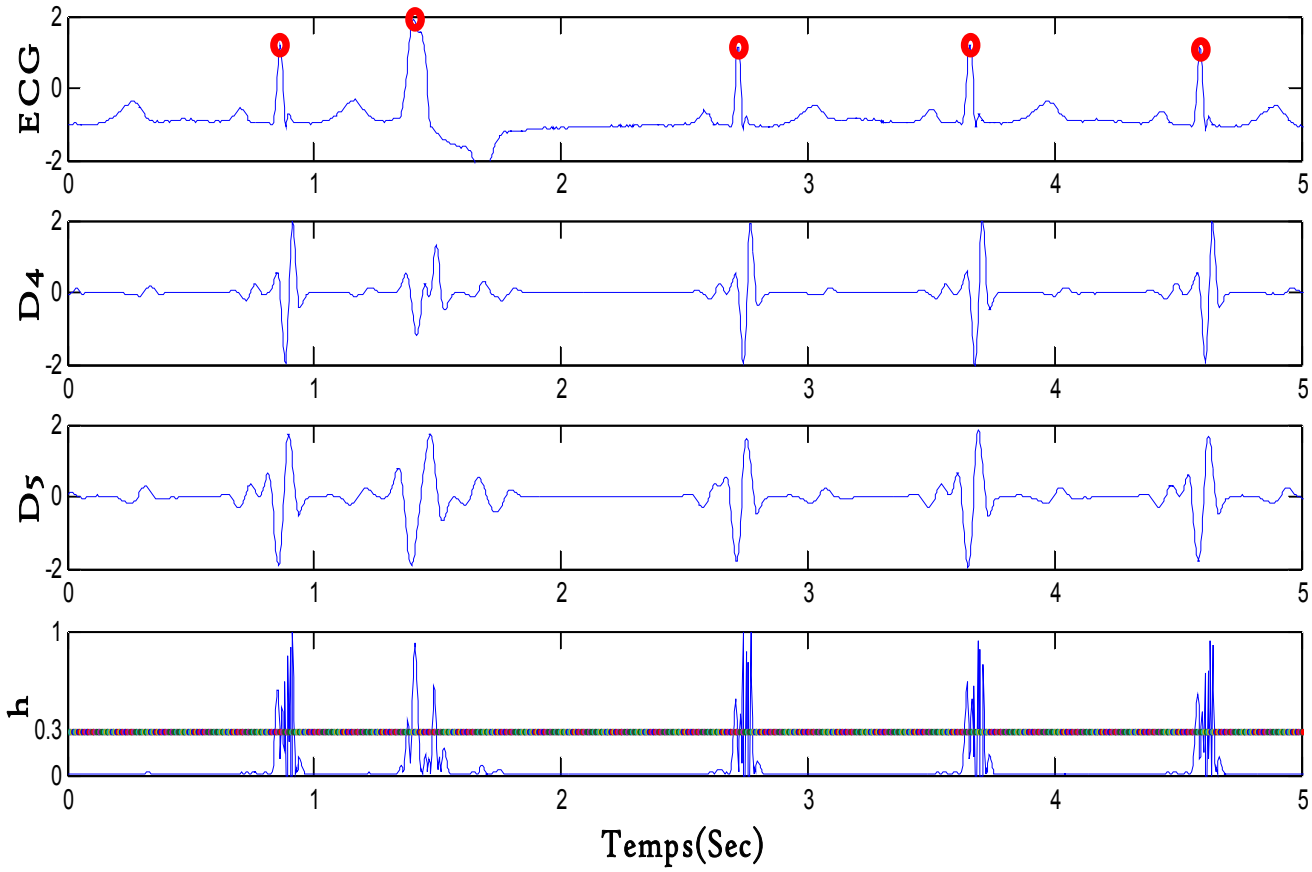


Figure III.10 Exemple de détection de pics R

III.7 Quantification des battements

Cette opération est communément appelée extraction des caractéristiques. Elle représente un volet qui doit être soigneusement réalisé en toute tâche de reconnaissance de formes. Le choix des paramètres discriminants est très important car un mauvais choix de ces paramètres veut dire une mauvaise description des échantillons et par conséquent un mauvais classifieur.

La quantification que nous envisageons ici consiste à représenter chaque battement segmenté par un vecteur dit vecteur de caractéristiques (attributs). Ce vecteur doit être optimal et les caractéristiques doivent être les plus discriminantes possibles sans s'éloigner de celles utilisées par les médecins. Après observation des enregistrements, ainsi que le contenu fréquentiel des différents types de battements, nous avons retenu les caractéristiques suivantes :

III.7.1 Energie du battement

La densité spectrale de puissance des deux types de battements illustrés sur la figure III.6 montre un décalage de fréquence entre eux. La décomposition du signal sur différents niveaux de résolution. Les battements normaux sont plus pertinents sur le détail d4 tandis que les ESV ont plus pertinents sur d5.

La capture de cette caractéristique est réalisée en calculant l'énergie totale de chaque segment QRS et ce, sur les détails d4 et d5 et sur le signal h.

Après la localisation des pics R, une fenêtre $w(n)$ de 180 ms est segmentée autour de la position de R. L'énergie du segment $w(n)$ est donnée par sa variance. Notons que ces segments sont forcement centrés (voir l'analyse multirésolution).

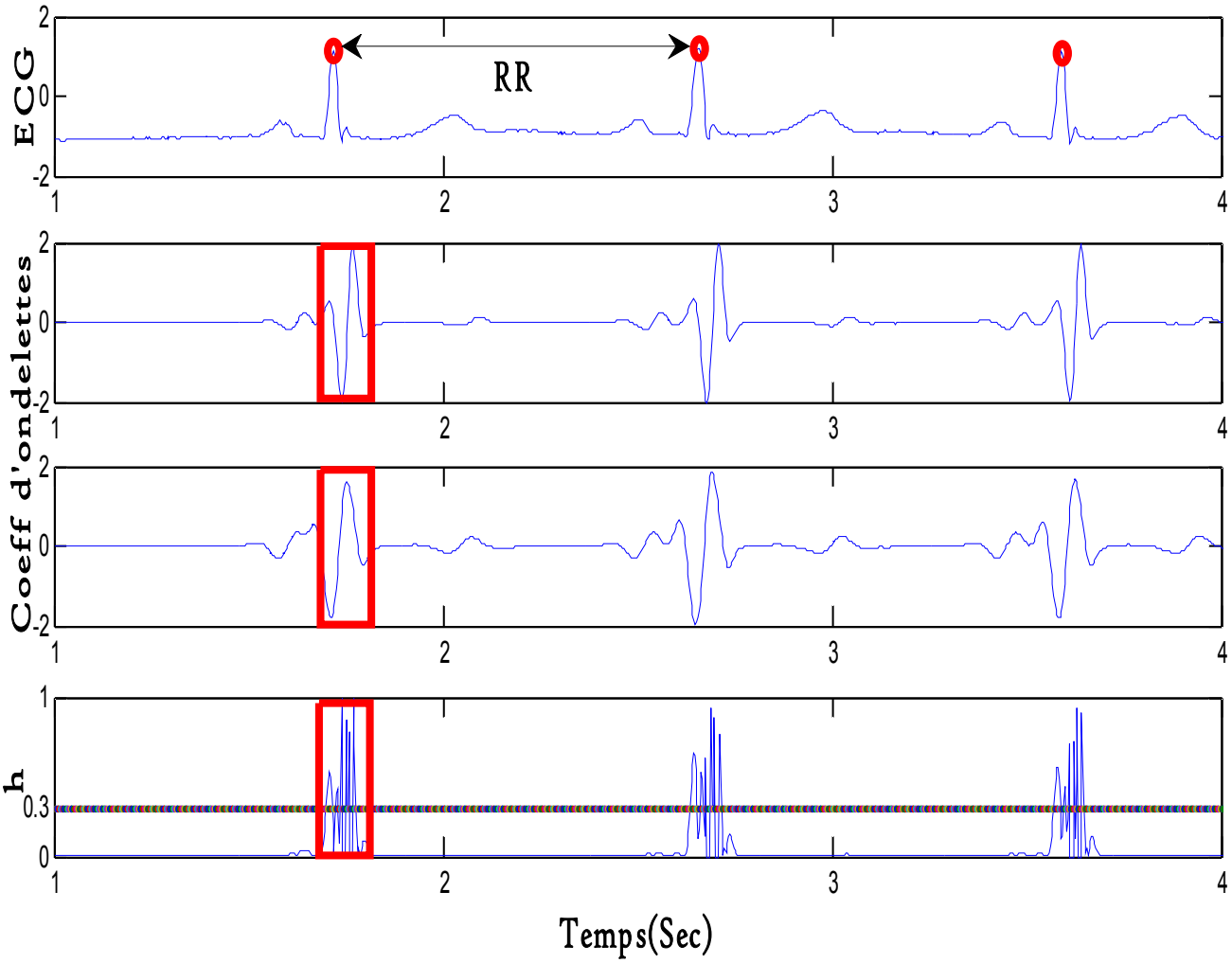


Figure III.11 Segmentation des QRS

III.7.2 Intervalles RR

L'absence de régularité des battements cardiaques est une caractéristique d'arythmie très importante. Nous avons constaté que les battements normaux sont réguliers tandis que les battements extrasystoliques surviennent de manière précoce et irrégulière (voir figures III.5 et III.10). L'irrégularité s'avère alors être un paramètre très discriminant.

La mesure de la régularité des battements est souvent effectuée par la mesure des intervalles *RR*. On appelle intervalle *RR* l'écart temporel entre le pic R du battement considéré et le pic R du battement précédent (voir figure III.9). Pour un battement *i*, on donne :

$$RR_i = R_i - R_{i-1} \tag{III.14}$$

III.7.3 Construction de la matrice de données

Après le calcul de caractéristique, le nombre de caractéristiques obtenues est de 4, c'est-à-dire que notre nuage de points aura P=4 attributs. Soit N le nombre de battements considéré. Les données sont alors disposées sous forme de matrice NxP (voir chap1).

$$\begin{pmatrix} x_1^1 & x_1^2 & x_1^3 & x_1^4 \\ .. & .. & .. & ... \\ x_i^1 & x_i^2 & x_i^3 & x_i^4 \\ .. & .. & .. & ... \\ x_N^1 & x_i^2 & x_i^3 & x_N^4 \end{pmatrix}$$

Pour un battement d'indice i :

$$\begin{cases} x_i^1 = var(d4_i) \\ x_i^2 = var(d5_i) \\ x_i^3 = var(d4.d5_i) \\ x_i^4 = RR_i \end{cases}$$

Pour s'assurer que les données obtenues ne soient pas bruitées et pour éliminer les attributs les moins discriminants, un prétraitement est nécessaire.

III.8 Etiquetage des battements

L'étiquetage correspond à l'attribution d'un label médical à chaque onde R définie lors de la segmentation. Nous définissons deux classes :

$$Classe\ positive\ (N_i) \rightarrow y_i = +1$$
$$Classe\ négative\ (ESV_i) \rightarrow y_i = -1$$

L'étiquetage repose entièrement sur les résultats de la segmentation et de la quantification, celles-ci doivent être réalisées avec beaucoup d'attention : plus la segmentation et la quantification sont pertinentes, plus l'étiquetage est simple et robuste.

III.9 Conclusion

Nous venons de voir les différentes étapes qui nous permettent de caractériser le signal ECG en ayant des connaissances et des notions sur les deux types de battements. Ce qui nous a permis d'extraire les paramètres discriminants et d'en faire un vecteur qui décrit au mieux l'exemple. Les vecteurs ainsi obtenus seront les entrées du classifieur SVM. L'ensemble de ces vecteurs est réparti en une base d'apprentissage et une base de test.

Les étapes à suivre pour atteindre notre objectif ainsi que les résultats obtenus à chaque étape de l'algorithme sont présentés dans le dernier chapitre.

Chapitre IV

Résultats et discussion

IV.1 Introduction

Dans ce chapitre, nous décrivons notre base d'étude ainsi que les enregistrements choisis puis les résultats obtenus avec les algorithmes que nous avons développés dans les chapitres précédents.

IV.2 Base de données

Les enregistrements ECG que nous avons utilisés dans ce travail proviennent de la base de données internationale MIT-BIH (Massachusetts Institute of Technologie/Beth Israel Hospital) arrhythmia data-base disponible sur le lien :
(www.physionet.org/physiobank/database).

Cette base est un ensemble de 48 enregistrements de 30 minutes chacun provenant d'un moniteur cardiaque Holter. Elle contient 23 enregistrements numérotés entre 100 et 124 pour le premier groupe ; et de 25 enregistrements numérotés entre 200 et 234 pour le deuxième groupe. Le premier groupe est prévu pour servir d'échantillon représentatif de variété de formes d'ondes qu'un détecteur d'arythmie pourrait rencontrer dans l'utilisation clinique courante ; tandis que le deuxième groupe est choisi pour inclure une variété de cas pathologiques. Les sujets étaient 25 hommes âgés de 32 à 89 ans, et 22 femmes âgées de 23 à 89 ans.

La base MITDB contient 116 137 battements annotés : chaque complexe QRS est décrit par une étiquette indiquant la position temporelle du pic R ainsi que le type de battement.
Chaque enregistrement est recueilli sur deux pistes correspondant aux dérivations D2 et V5.

Les signaux sont échantillonnés avec une fréquence de 360 Hz. Les enregistrements contiennent plusieurs bruits, artefacts, différents types de battements : ventriculaire, supra-ventriculaires, jonctionnels et plusieurs autres anomalies de conduction, c'est pourquoi, elle est très souvent utilisée pour valider tout algorithme lié aux arythmies cardiaques et aux artefacts.

IV.2.1 Signaux choisis

Pour valider notre algorithme, nous avons choisi des enregistrements contenant le plus d'extrasystoles et quasiment deux classes de battements.

a. L'enregistrement 208

Cet enregistrement est l'ECG d'une femme âgée de 23 ans, électrode V1, groupe II. Nous l'avons choisi car il possède un grand nombre de battements extrasystoliques uniformes. Cet enregistrement contient 2955 battements dont 1586 des battements sont normaux et le reste des battements est considéré comme extrasystoliques. La figure IV.1 montre un segment de cet enregistrement.

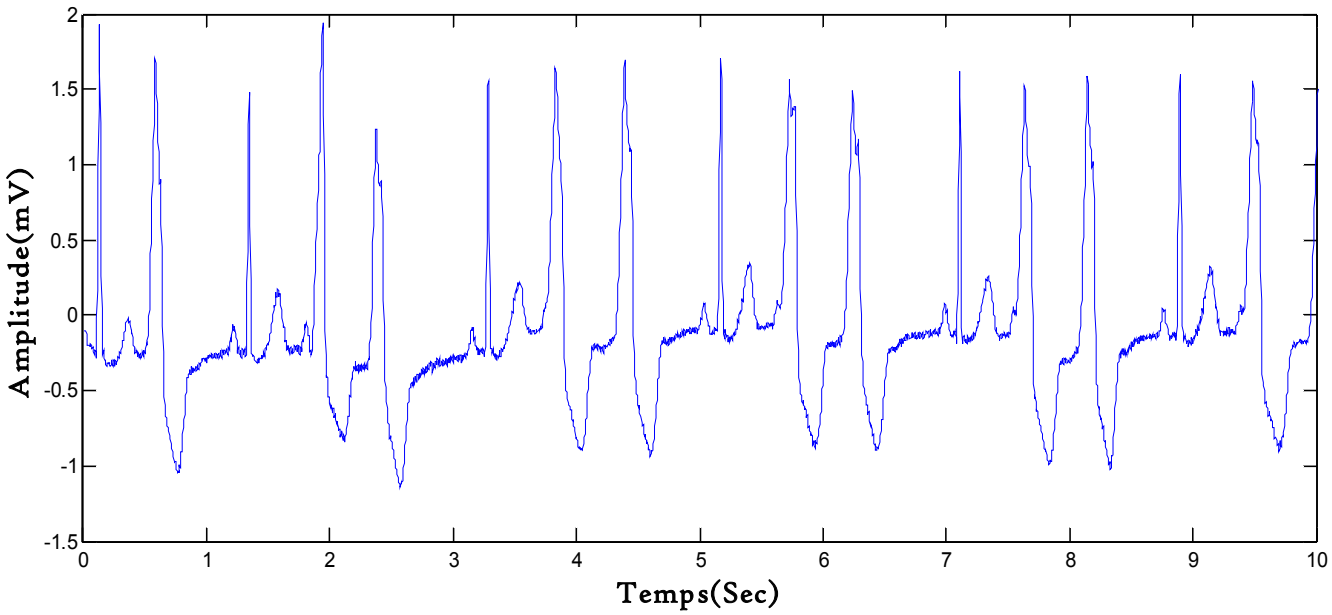


Figure IV.1 : Segment de l'enregistrement 208.

b. L'enregistrement 106

L'enregistrement 106 représente l'ECG d'une femme âgée de 24 ans provenant de la dérivation V1, groupe I. Il est choisi car les extrasystoles sont multiformes. Cet enregistrement contient 2027 battements dont 1507 sont considérés comme étant normaux et 520 comme étant extrasystoliques, ici les PVC sont multiformes comme l'indique la figure IV.2.

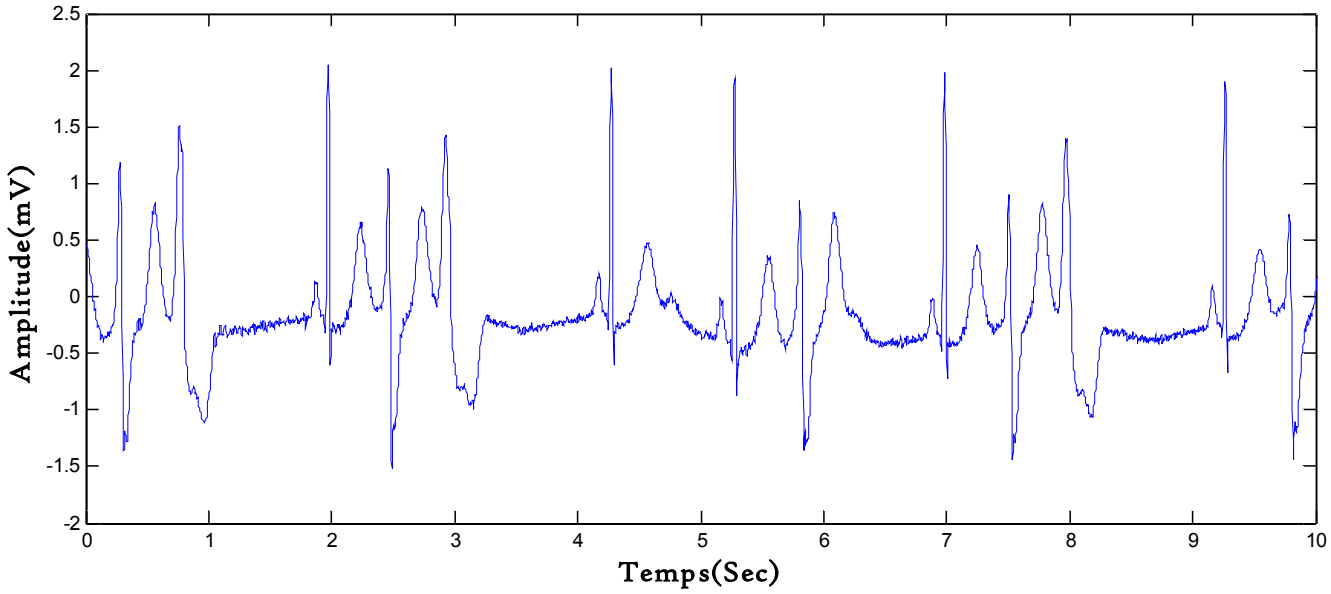


Figure IV.2 : Segment de l'enregistrement 106.

c. L'enregistrement 119

L'enregistrement 119, a été effectué sur une femme âgée de 51 ans, dérivation V1, groupe I. Dans ce cas, les extrasystoles sont uniformes. Cet enregistrement comporte 1987 battements dont 1543 sont normaux et 444 sont des ESV (voir figure IV.3).

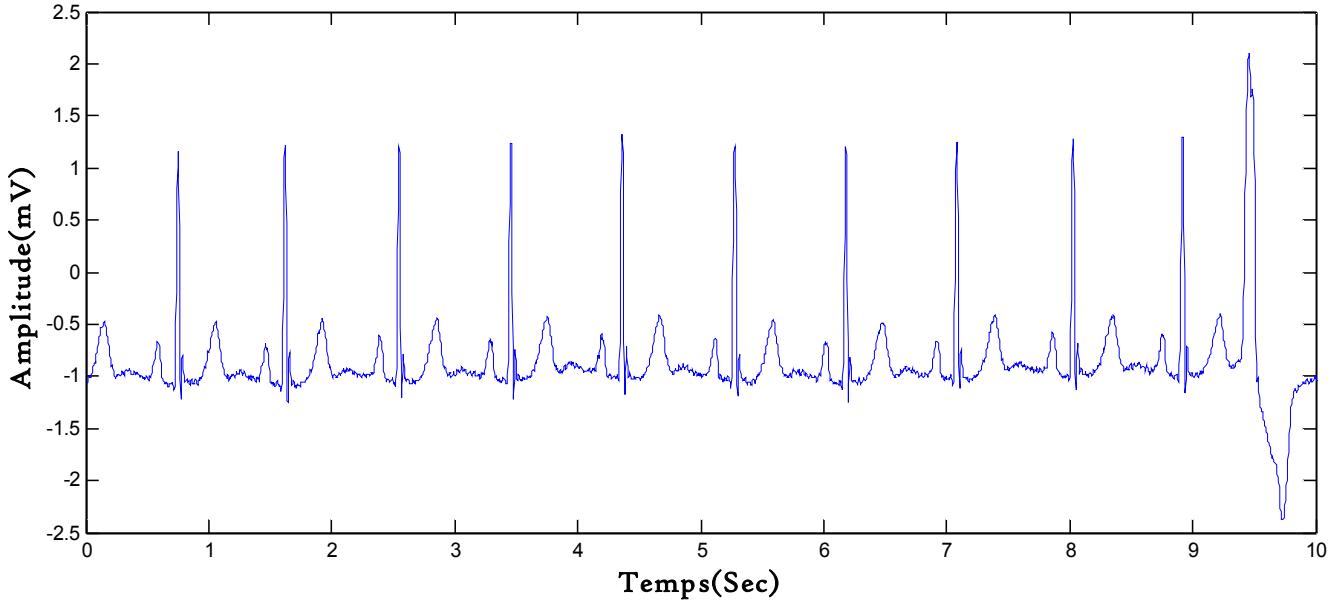


Figure IV.3 : Segment de l'enregistrement 119.

Nous avons choisi ces enregistrements en particulier car ils sont différents et pour en faire une base de données riche en information. Ceci nous permettra d'entrainer notre classifieur pour reconnaitre différentes formes d'ESV, ceci donnera un classifieur global. Sur ces enregistrements choisis, nous allons extraire une base d'apprentissage et une base de test pour le SVM.

IV.3 Organigramme général

Les données ainsi collectées subissent d'abord un prétraitement afin de :

- Supprimer les ondulations de la ligne de base
- Supprimer les bruits de haute fréquence.

En utilisant différents niveaux de résolution, les QRS sont détectés puis segmentés et enfin quantifiés pour engendrer des vecteurs de caractéristiques qui servent pour l'apprentissage et les tests. Ceci est résumé par l'organigramme de la figure IV.4.

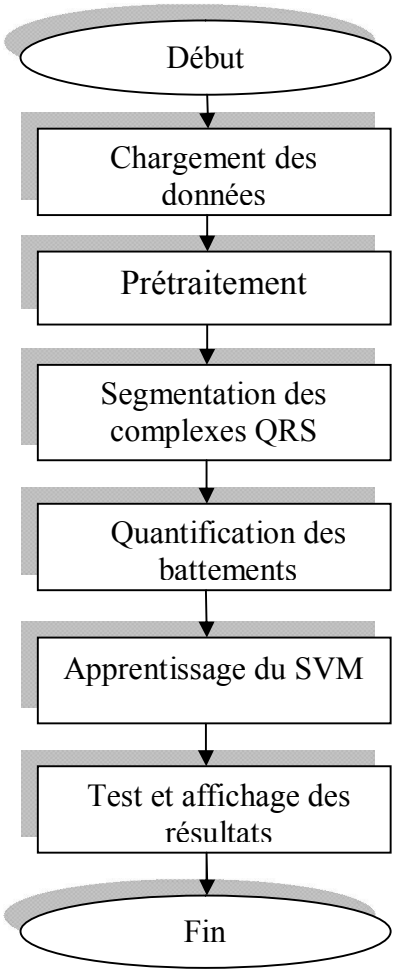


Figure IV.4 : Organigramme global

IV.4 Quelques résultats de localisation des pics R

L'analyse multirésolution est l'outil idéal pour la séparation de ces deux types de battement, en séparant les hautes des basses fréquences, les deux types de battements peuvent être analysé séparément. Quant au choix de l'ondelette mère, nous avons choisi l'ondelette de Haar car elle possède de bonnes propriétés de localisation temporelle**.**

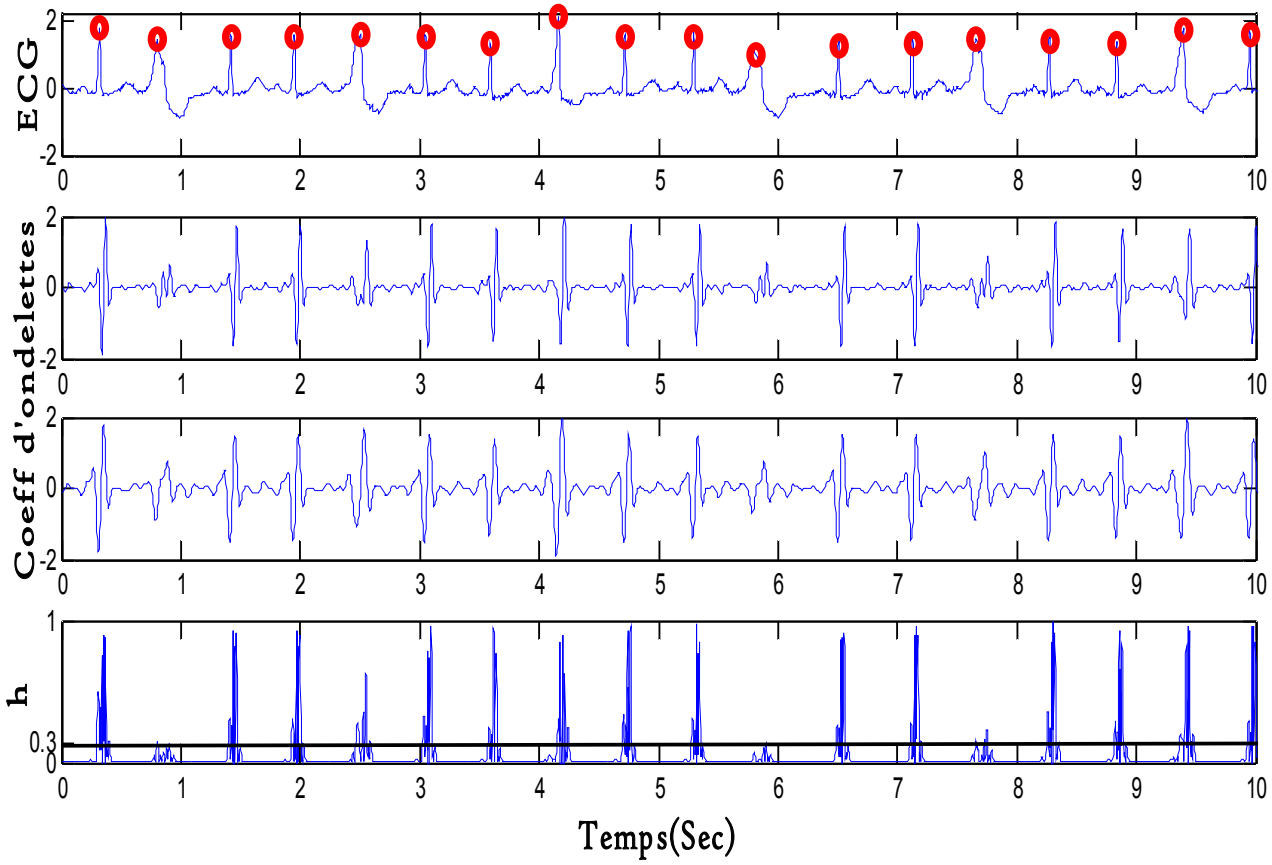
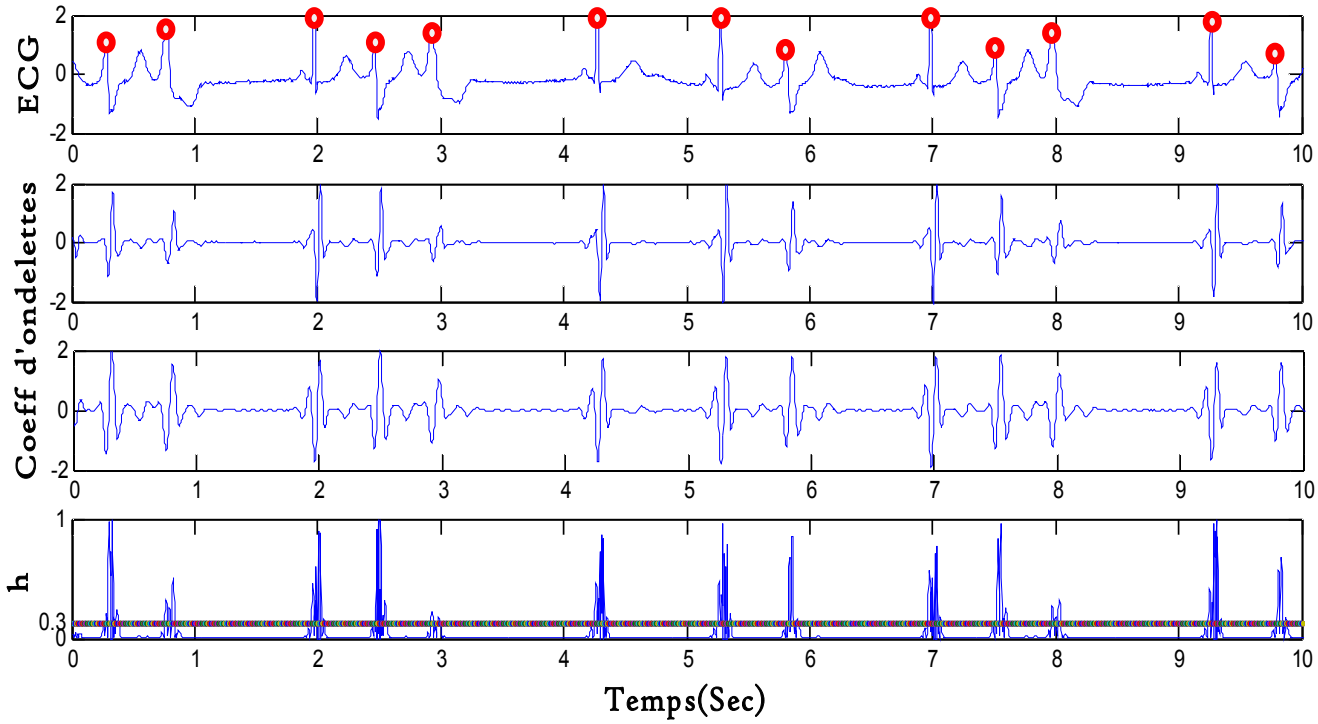
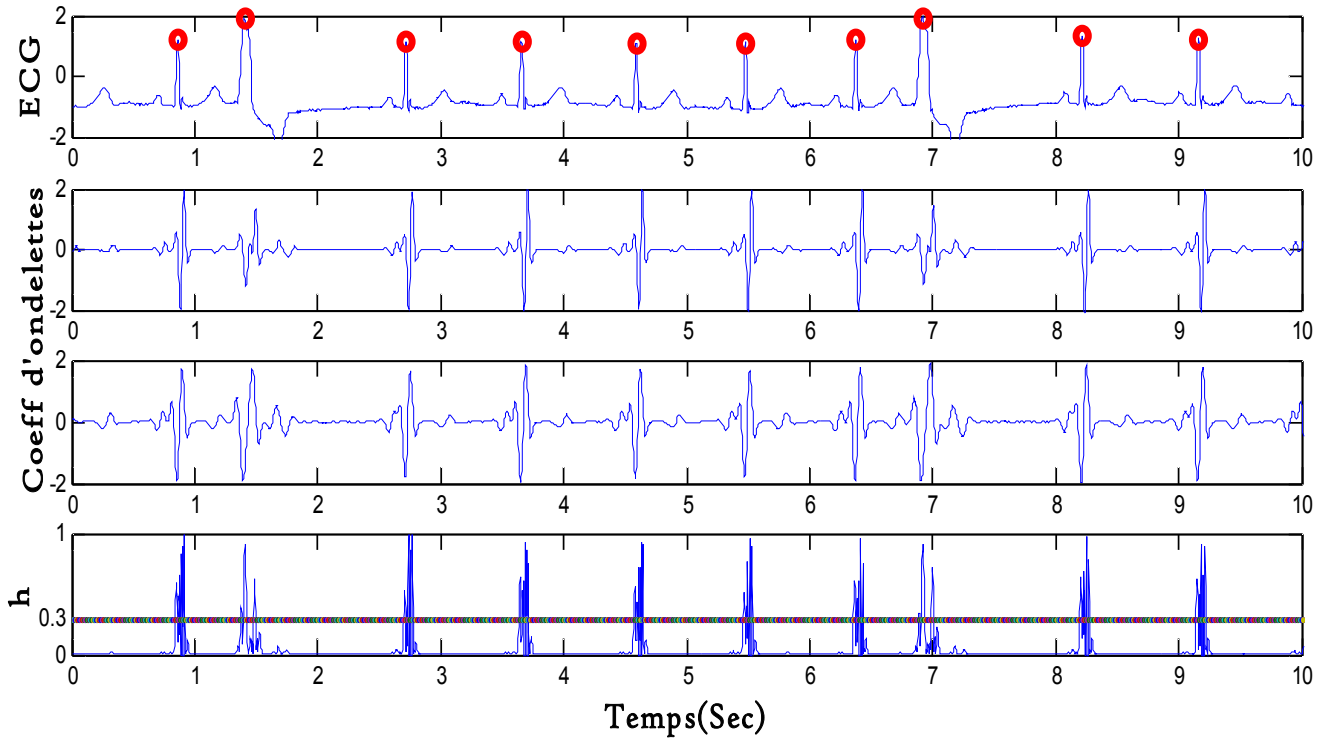


Figure IV.5 : Exemple de décomposition et de détection sur un segment du signal 208

Nous remarquons sur les figures IV.5, IV.6 et IV.7 que les niveaux 1, 2 et 3 ([22.5-180]) ne sont pas considérés car ils sont en dehors de la bande d'intérêt. Les battements normaux sont plus significatifs sur le niveau de résolution 4 [11.25 Hz, 22.5 Hz] tandis que les battements ESV sont plus pertinents sur le niveau 5 [5.62 Hz, 11.25 Hz]. Tous les pics sont correctement localisés malgré que certains battements sont atypiques.



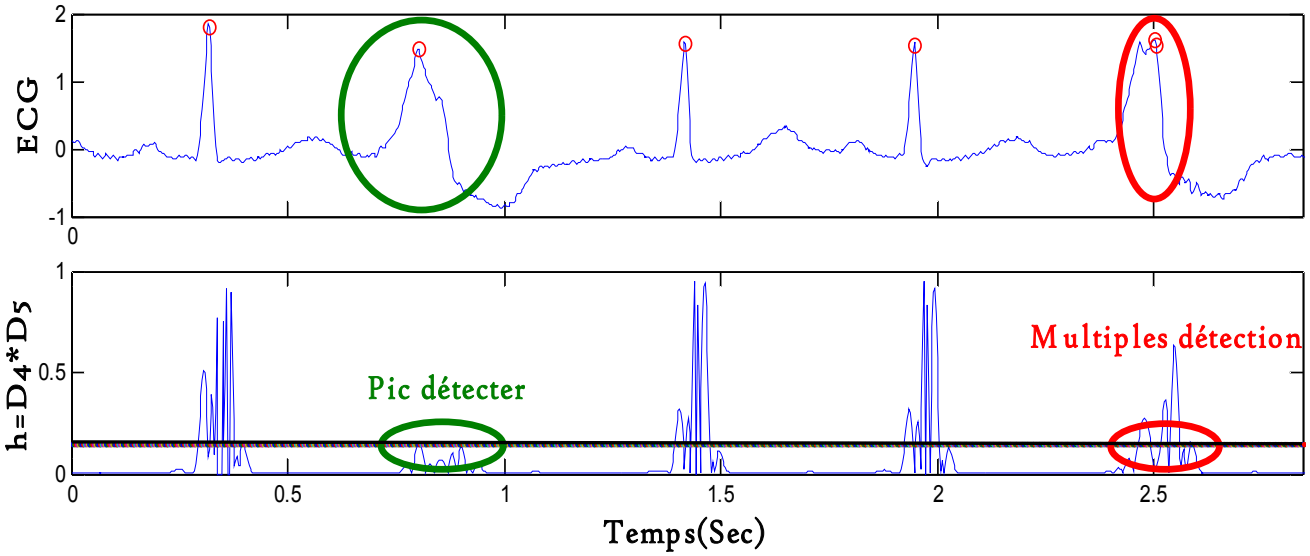
IV.6 : Exemple de décomposition et de détection sur un segment du signal 106



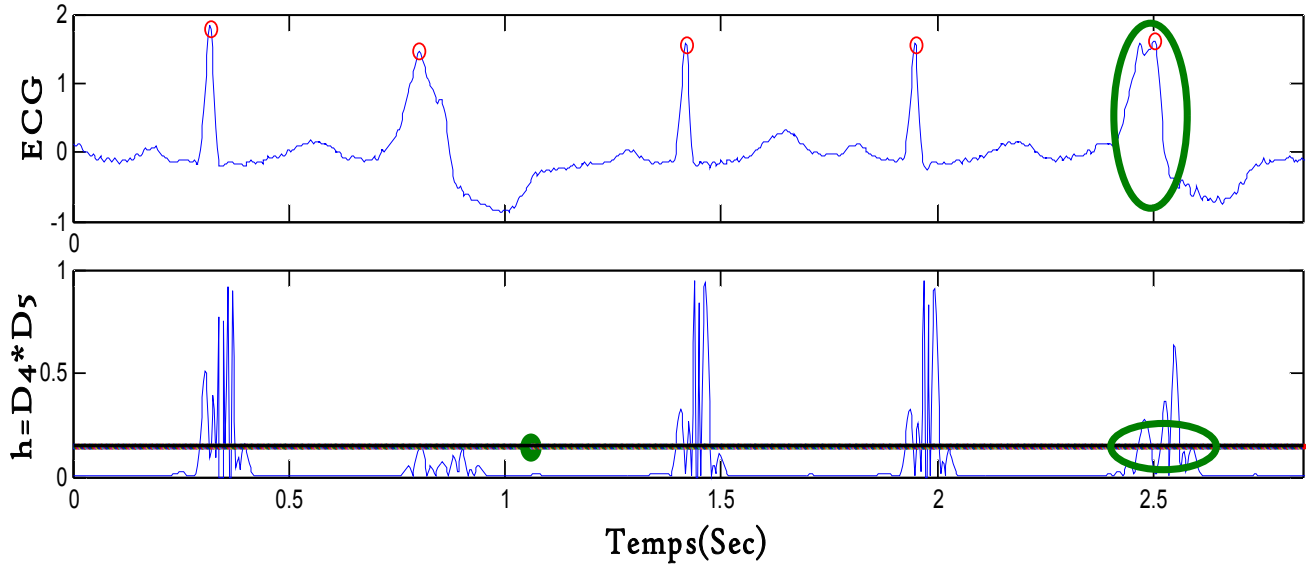
IV.7 : Exemple de décomposition et de détection sur un segment du signal 119

Remarque

La détection des battements normaux ainsi que la majorité des ESV ne pose aucun problème avec un seuil de 30%. Dans certaines ESV atypiques et de faible amplitude, une recherche de battements omis a du être effectuée comme il a été bien indiqué dans l'algorithme de détection évoqué au chapitre III. Dans le cas de battements trop larges ou encore ceux suivis d'une onde T très important, des détections multiples ont été observées. La correction est également prévue dans l'algorithme. Un exemple est donné en figure IV.8.



IV.8 : Exemple de détection omise ou multiples



IV.9 : Correction des détections omises ou multiples.

Par précaution et afin que l'onde T ne soit pas considérée comme un QRS, une détection dans un intervalle de 200ms (période réfractaire) sera supprimée.

Avec cet algorithme, nous avons essayé de contourner tous les problèmes qui peuvent s'opposer à notre objectif qui est la détection des pics R. Ainsi Le pourcentage de bonne détection est de 100%.

IV.5 Résultat de la quantification

Une fois les pics détectés, il nous reste plus qu'à trouver les paramètres les plus discriminants pour transformer notre signal en une matrice qui le traduit. Cette phase de travail repose entièrement sur l'utilisateur, car c'est à lui de trouver ces paramètres. Nous avons remarqué que les deux signaux n'appartienne pas a la même bande de fréquence c'est ce qui nous à conduits à utiliser le contenu fréquentiel comme paramètre. Les deux niveaux de résolution, l'amplitude de leurs produits peuvent aussi être pris comme paramètres de caractérisation.

Nous remarquons aussi que la distance entre deux pics R est régulière lorsque c'est un battement normal et irrégulier lorsqu'il s'agit de battement ESV, cette distance aussi peut être utilisée pour caractériser les deux types de battement.

Nous obtiendrons ainsi pour chaque battement un vecteur de quatre attributs, il aura la forme suivante :

*variance (d4.*d5)*	*variance (d4)*	*variance (d5)*	*RR(i)*
0,3952	0,1000	0,2436	174
0,2005	0,5387	0,6602	216
0,1952	0,5423	0,6472	191
0,7295	0,2698	0,7166	199
……..	……..	……..	…..

Tableau 2 : Quelques échantillons de la matrice obtenue

Pour la prochaine étape de ce travail qui est la classification de données, nous aurons besoin de représenter le résultat de la classification en deux dimensions des battements normaux et ceux extrasystolique, pour cela, réduire la taille des attributs est indispensable, l'ACP est l'outil qui accomplira cette tâche.

Notre matrice de départ deviendra :

0,0600	-1,2122	-1,4855	-1,2031
-0,7160	0,8121	0,4287	1,2031
-0,7370	0,8288	0,3689	-0,2291
1,3930	-0,4287	0,6878	0,2291
……..	……....	……	……

Tableau 3 : Matrice normalisée.

+2,2380	+1,0441	+0,1684	0
-1,8720	-0,0137	+0,5467	0
-1,2900	+0,6370	-0,5905	0
+0,9240	-1,6674	-0,1246	0
……..	……....	……	……

Tableau 4 : Données projetées sur les axes artificiels par une ACP.

Notons qu'à chaque ligne de la matrice, c'est-à-dire à chaque vecteur ou échantillon, un label a été affecté ***car il s'agit d'un classifieur supervisé.***

IV.6 Classification des données obtenues

IV.6.1 Introduction

Cette étape met en évidence l'objectif même de ce travail, la classification binaire de battements cardiaque avec SVM. Comme nous l'avons déjà vu dans le chapitre II, le SVM sépare les données tous en veillant que celles-ci soit largement bien classées.

IV.6.2 Protocole expérimental

- ***Base d'apprentissage***

Pour constituer une bonne base d'apprentissage qui nous permettrait la généralisation de notre classifieur et d'obtenir ainsi un classifieur global, nous avons pris 50 battements de chaque enregistrement soit 150 battements étiquetés par des cardiologues.

Base de test

Contient 1450 battements pris par l'ensemble des 3 enregistrements. Cette base est complètement dissociée de la base d'apprentissage dans le but d'évaluer la capacité de généralisation de notre algorithme de classification.

Réglage des hyper- paramètres

Dans le cas des données linéairement séparable, SVM a la capacité de tolérer les erreurs de classement (marge souple) toute en les pénalisant, le degré de pénalisation est défini par l'utilisateur selon le domaine ou l'application par le paramètre **C** de tolérance aux erreurs. Plus **C** est important, plus la tolérance aux erreurs est minime et vice-versa. Le coût **C** de l'erreur peut être asymétrique, c'est-à-dire, affecter $\boldsymbol{C_p}$ pour la classe positive et $\boldsymbol{C_n}$ pour la classe négative**.**

Dans le cas des données non linéairement séparables, l'utilisation des fonctions noyaux pour passer à l'espace de rediscription est indispensable, le noyau gaussien a un paramètre $\boldsymbol{\sigma}$ pour contrôler la largeur de la gaussienne, en choisissant ce paramètre nous choisissons le degré de similarité entre les exemples (plus ce paramètre est grand plus la similarité entre les exemples augmente).

Ces choix sont le plus souvent faits par une technique de validation croisée, dans laquelle on estime la performance du système en la mesurant sur des exemples n'ayant pas été utilisés en cours d'apprentissage.

Validation croisée

La validation croisée est la méthode la plus populaire de réglage des hyper-paramètres utilisés dans les méthodes d'apprentissage. Cela consiste à discrétiser les espaces de chacun des hyper-paramètres à régler puis de tester successivement toutes les combinaisons possibles et définie sur une échelle logarithmique.

L'apprentissage s'effectue sur un sous-ensemble de la base et le test sur les données restantes, que l'on appelle ensemble de validation. Le classifieur choisi est à son tour évalué sur un ensemble de test indépendant

Nous avons utilisé un noyau gaussien et un rapport de paramètre de pénalisation des deux classes $\frac{C_p}{C_n} = 1$ (coût symétrique). Nous avons choisi ces deux paramètres en utilisons la technique de validation croisée et nous avons obtenu le résultat suivant :

Le paramètre de pénalisation C=10 et σ pour le noyau gaussien de 0.9.

IV.6.3 Résultats obtenus

Apprentissage

Nous avons obtenu un nombre de 29 vecteurs de support parmi les 150 échantillons choisis pour la base d'apprentissage.

L'apprentissage par SVM revient à chercher les deux paramètres qui sont w et b.

Le vecteur ***w*** possédant un nombre de composantes égal à la dimension de l'espace d'entrée et qui désigne la direction de l'hyperplan aura cette valeur :

$$w = [-4.4832 \quad 16.4492 \quad 0.1438 \quad 6.4875]$$

Quant au biais b qui est le décalage par rapport à l'origine a la valeur suivante :

$$b = 0{,}31$$

Test

Nous avons testé les 1450 battements de la base de test et le résultat de la classification est représenté sur la figure IV.10 ; la projection sur un espace à deux dimensions nous permet de visualiser le résultat de la classification de façon effective. Nous remarquons sur la figure que les deux classes sont bien séparées. Chaque classe se présente cependant sous forme de sous groupes d'échantillons. Ceci est dû au fait que les battements proviennent de différents sujets. L'avantage est de monter que le classifieur est global.

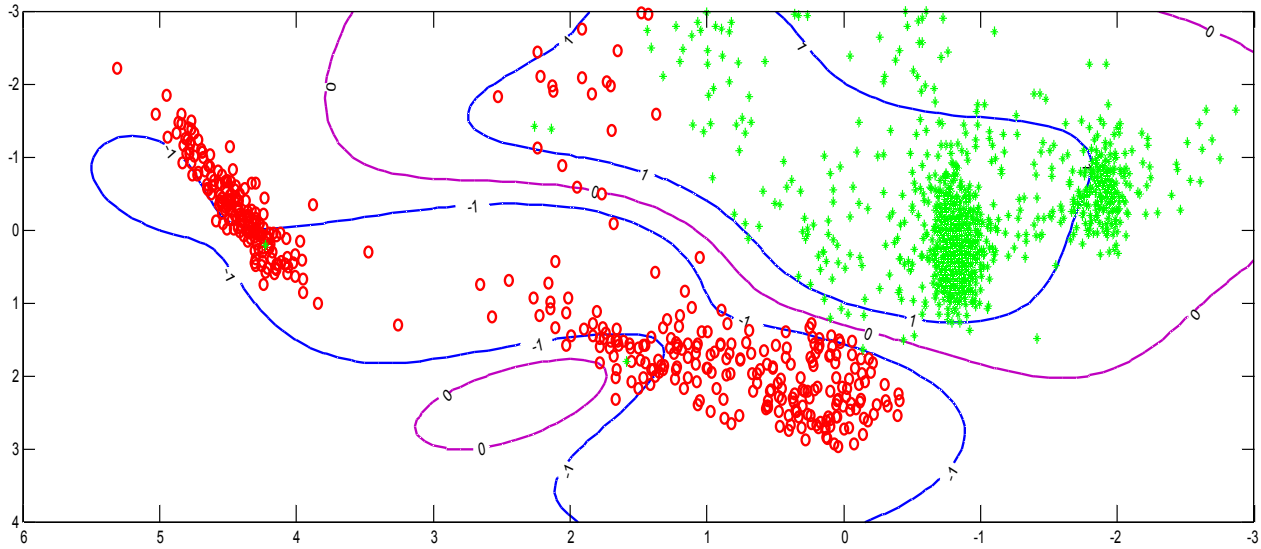


Figure IV.10 : Résultat de la classification (classe positive : étoiles vertes; classe négative ronds rouges).

IV.6.4 Performances du classifieur

Tout algorithme de reconnaissance de forme doit être évalué par des paramètres statistiques qui permettent de discuter ses performances et de le comparer à d'autres algorithmes. Souvent les paramètres statistiques suivants sont utilisés.

- ***Le taux d'erreurs***

Le taux d'erreurs est donné par la formule suivante :

$$E_r = \frac{FP + FN}{N}$$

Ou FP désigne les faux positifs, FN les faux négatifs et N le nombre total d'échantillons. Nous avons obtenu un résultat de 3.24%.

Le taux de classifications correctes est :

$$C_c = \frac{VP + VN}{N}$$

Ou VP désigne les vrais positifs, VN les vrais négatifs. Nous avons obtenu un résultat de 96.75%.

IV.7 Conclusion

Nous venons d'afficher les résultats de la classification des deux types de battements en expliquant chaque étape de l'algorithme. Les résultats obtenus sont assez satisfaisants mais peuvent bien sur être améliorés. A travers ces deux volets à savoir l'extraction des caractéristiques et le modèle de classifieur obtenu.

Conclusion Générale

Conclusion générale

Ce mémoire s'inscrit dans le domaine de la reconnaissance de formes. Un domaine à la croisée de plusieurs disciplines traitement de l'information et les mathématiques appliquées, notamment l'optimisation.

Sur le plan pédagogique, nous avons saisi cette opportunité afin d'approfondir nos connaissances acquises pendant le cursus universitaire et ce, en traitement du signal, en caractérisation et classification de données.

Nous pouvons dire aussi que ce travail est une initiation à la recherche car il nous a permis d'apporter une petite contribution originale à un domaine de recherche très actif. Nous avons relativement appris à choisir un support de travail, comment choisir un outil mathématique et comment évaluer et interpréter un résultat.

Le premier volet du travail a consisté à localiser, segmenter et caractériser les battements cardiaques. L'algorithme utilisé pour la localisation a été jugé très performant même dans le cas de battements atypiques. La caractérisation des battements a été faite dans le domaine temporel pour l'étude de régularité et dans le domaine temps-échelle pour extraire l'information fréquentielle et énergétique.

Le deuxième volet du travail comprend le fondement mathématique des SVMs. Ce classifieur a fait ses preuves en termes de performances en discrimination et en pouvoir de généralisation.

Les résultats obtenus sont jugés assez satisfaisants. La localisation automatique des pics R a été de 100% sur tous les segments utilisés. Après leur caractérisation et leur classification, le SVM a affiché un taux de bonne classification de 96.75%.

Comme nous l'avons évoqué plus haut, il s'agit d'un domaine de recherche très actif. Ce travail reste bien sur ouvert pour toute amélioration. Des améliorations peuvent êtres apportées sur le premier volet en ajoutant par exemple d'autres caractéristiques, ou sur le classifieur en choisissant par exemple la base d'apprentissage de façon plus rigoureuse. Le choix des échantillons pour former la base d'apprentissage est important pour avoir un bon coût de classification, le nombre d'exemples doit être élevé (apprentissage par cœur), sinon les échantillons qui forment la base d'apprentissage doivent être éloignés autrement dit c'est un apprentissage de différents cas possibles.

Bibliographie

Bibliographie

[1] AMIROU Ahmed. *Optimisation des SVMs pour la discrimination de signaux*. Thèse de doctorat d'université. Tizi Ouzou : Université Mouloud Mammeri. 2015. P.1-11.

[2] ZIDELMAL Zahia. *Caractérisation et classification de données*. Cours. Tizi-Ouzou : Université Mouloud Mammeri. 2015.

[3] Abdelhamid DJEFFAL. *Fouille de données avancée*. Cours. Biskra : Université Mohamed Khider , 2014/2015, P.1-7

[4] MAHDJANE Karima. *Détection d'anomalies sur des données biologiques par SVM*. Thèse de Magister. Electronique option télédétection. Tizi-Ouzou : Université Mouloud Mammeri. 2012. P.6, 7,11

[5] MARREF Nadia. *Apprentissage Incrémental & Machines à Vecteurs Supports*. Mémoire de fin d'études. Informatique industrielle. BATNA : Université HADJ LAKHDAR, 2013, P.19-24,P.51-56

[6] LAMICHE Chaabane. *Fusion et fouille de données guidées par les Connaissances : application à l'analyse d'image*. Thèse de doctorat d'université. BISKRA : Université MOHAMED KHIDER, 2013, P.15-21

[7] ZIDELMAL Zahia. *Reconnaissance d'arythmies cardiaque par Support Vector Machines*. Thèse de Doctorat, Université Mouloud Mammeri-Tizi Ouzou. 2012. P.

[8] V. Vapnik. *Statistical Learning Theory*. New York , USA, 1998.

[9] B. Scholkopf and A. Smola. *Leaning with Kernels*. MIT Press, 2001.

[10] John C.platt, Sequentiel Minimal Optimization : A Fast Algorithm for Training Support Vector Machines, technical report MSR-TR-98-14, 1998.

[11] Thorsten Joachims, *SVM light Support Vector Machine*, technical report, cornell University, 2002.

[12] S.V.N.Vishwanathan & M.Narasimha Murty, *A Simple SVM Algorithm*, technical report, Dept. of CSA, Indian Institute of Science, Bengalore, India, 2012.

[13] Thomas G. Dietterich and G.Bakiri. *Solving multiclass learning problems via error-correcting output codes"*. Journal of Artificial Intelligence Research, 2 :pages 263–286, 1995.

[14] J.COGET. *Physiologie du neurone*. Support de cour. Lille : Université de Lille. 2009